1 Introduction

1.1 Background

In October 1992, the INCOSE formed the Capability Assessment Working Group (CAWG). The CAWG charter included developing “a method for assessing and improving the efficiency and effectiveness of systems engineering” and one of its goals was to develop a Systems Engineering Capability Assessment Model (SECAM). Generation of the INCOSE SECAM was begun in November 1993 and completed as an initial release, Version 1.00, in February 1994. Version 1.00 of the SECAM Assessment Method (then called Supporting Documents) was released in March 1994. Following the initial release, five major updates and several minor updates were made to the INCOSE SECAM and several updates were made to the SECAM Assessment Method. The current release of the INCOSE SECAM is Version 1.50, dated June 1996; the current release of the SECAM Assessment Method is Version 1.50, dated July 1997.

In January 1994, the EPIC (then called Industrial Collaboration) began generating the Capability Maturity Model (CMM) for Systems Engineering (SE-CMM) and completed it as an initial release, Version 1.0, in December 1994. Version 1.0 of the SE-CMM Appraisal Method was released in June 1995. Following the initial release, one update was made to both the EPIC SE-CMM and the SE-CMM Appraisal Method. The current release of the EPIC SE-CMM is Version 1.1, dated November 1995; the current release of the SE-CMM Appraisal Method is Version 1.1, dated 1996.

In March 1996, an effort was initiated under the auspices of the EIA G-47 (Systems Engineering) Committee to merge the current versions of the INCOSE SECAM and its Assessment Method with the current versions of the EPIC SE-CMM and its Appraisal Method. The resulting EIA Systems Engineering Capability Model (SECM) and SECM Appraisal Method will be proposed as a US national standard for the measurement and improvement of systems engineering capability.

1.2 Benefits Associated with the Use of this Interim Standard

Proper implementation of this Interim Standard is intended to improve the capability to perform systems engineering. Improved capability enables an organization to:

a) Reduce cycle time from concept to deployed system products;
b) Improve the match of deployed system product capability with stakeholder requirements;
c) Reduce total ownership cost of system products;
d) Reduce the number of engineering changes;
e) Improve system quality;
f) Improve communications among personnel involved in the engineering of a system;
g) Improve ability to sustain and upgrade system products after deployment; and
h) Reduce development risks.

By proper implementation, it is meant that:

a) Processes, activities, and tasks of this Interim Standard are appropriately tailored (see Annex A and EIA/IS 731-2, *EIA SECM Appraisal Method*);
b) Skilled personnel are used to accomplish the purpose of this Interim Standard as tailored;
c) Users of this Interim Standard have training and familiarity with the usage of this Interim Standard.

2 Scope

2.1 Purpose

The purpose of this Interim Standard is to support the development and improvement of systems engineering capability.

2.2 Coverage

The scope of this standard includes all activities that associate with or enable systems engineering. Systems engineering is an inter-disciplinary approach and means to enable the realization of successful systems. In this context, systems engineering is not limited to what either Systems Engineering organizations or Systems Engineers do. Rather it is the interaction of many people, processes, and organizations resulting in the accomplishment of the required activities.

This Interim Standard is intended to provide complete coverage of EIA 632, *Processes for Engineering a System*, and be consistent with both it and Institute of Electrical and Electronic Engineers (IEEE) 1220-1994, *IEEE Trial-Use Standard for Application and Management of the Systems Engineering Process*.

2.3 Application

The EIA SECM, EIA 731-1, is to be used in conjunction with the SECM Appraisal Method, EIA 731-2, to develop, improve, and assess systems engineering capability.

This Interim Standard applies to programs and organizations doing systems engineering: small or large; simple or complex; software intensive or not; precedented or unprecedented. It applies to systems that may contain hardware, software, personnel, facilities, data, materials, services, or techniques. This Interim Standard is applicable to the engineering of a new system or the reengineering of a legacy system, or portions thereof.

This Interim Standard is intended solely to be used for self-development, self-improvement, and self-appraisal. Organizations should not apply this Interim Standard to suppliers as a means of source selection or as a means of qualification to be a supplier.

2.4 Limitations

This Interim Standard is not intended to specify the details of “how to” implement process activities. It does not specify the methods or tools a developer would use to accomplish required activities. The developer **will** select methods, techniques, and tools that are consistent with program or organization needs, directives, and procedures.

Adherence to this Standard **shall** be entirely voluntary and within the discretion of individual organizations.

This Interim Standard does not prescribe the name, format, content, or structure of documentation. Throughout this Interim Standard, the terms “document” or “documentation” are used to mean a collection of data regardless of its medium.

This Interim Standard is, to a large extent, a process-based systems engineering capability model. Process maturity indicators were developed first because they were deemed easiest to develop and had received the most attention in other efforts, e.g., ISO initiatives, and other disciplines, e.g., software. This Interim Standard also includes non-process indicators of systems engineering capability. These non-process indicators represent high leverage characteristics of systems engineering capability.

This Interim Standard has not been validated. An inherent, necessary component of validation would be to demonstrate quantitatively that improvement in systems engineering capability, as measured by appraisal results, can lead to a more effective organization. This will be reflected by systems engineering's impact on product development in terms of the "bottom line", such as quicker time to market, lower product cost, reduction in cycle time, etc. Many organizations have utilized the INCOSE or EPIC models and have performed assessments using the applicable methodologies. This standard builds on the experience gained from these models that were deemed to be of value to the organizations that have been assessed.

2.5 Tailoring

This Interim Standard contains a set of Focus Areas, Themes, Practices, and Attributes designed to be tailored. Tailoring (see Annex A and EIA 731-2 for guidance) is deletion of non-applicable Focus Areas, Themes, Practices, and Attributes, or addition of unique or special Focus Areas, Themes, Practices, and Attributes provided in organization policies and procedures, or in an acquirer-supplier agreement.

3 Normative References

EIA 632, *Processes for Engineering a System*

EIA STD 649, *National Consensus Standard for Configuration Management*

IEEE STD 1220 - 1994, *IEEE Trial-Use Standard for Application and Management of the Systems Engineering Process*

4 Model Definition

4.1 Using the Model

The SECM is structured to support a wide variety of improvement activities including appraisals, process improvement, and process design. A short description of each is provided here.

Appraisal:

Appraisals may be either self-administered or augmented by expert “facilitators” from inside or outside the organization. This model is intended for internal process improvement. It has not been calibrated or qualified for evaluation of a potential vendor’s capability to perform its systems engineering process. The latter use, vendor selection, could lead to incorrect conclusions about a supplier’s ability to perform, and is not recommended.

Although it is not required that any particular appraisal method be used with the SECM, an appraisal method designed to maximize the utility of the model has been designed by the SECM Project. Part 2 of this Interim Standard, designated EIA 731-2 SECM Appraisal Method, fully describes the recommended method and includes support materials.

Process Improvement:

Either with or without an appraisal to benchmark an organization’s systems engineering practices, an organization may use the SECM for guidance to design an improvement program. There are many sources within industry for approaches to organizational improvement and most should be able to be used with the SECM or adapted for SECM use.

A capability level and improvement method should be developed for each Focus Area, independently of each other. Any process improvement effort, using any reference model, should be constructed to support the business goals of the organization. An organization using the SECM should prioritize the Focus Areas relative to their business goals and strive for improvement in the highest priority Focus Areas first.

Process Design:

The first step in designing processes that will meet the business needs of an enterprise is to understand the business, product, and organizational context that will be present when the process is being implemented. Some questions that need to be answered before the SECM 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 programs?
- How are support functions handled (e.g., by the program or the organization)?
- What are the management and practitioner roles used in this organization?
- How critical are these processes to organizational success?

An organization must have a stable baseline to determine whether future changes constitute improvements. Appraisals can help in the establishment of such a baseline.

Common sense dictates that there is no value in looking for improvements in a process that the organization does not perform. The model may be tailored to an enterprise’s or program’s needs by selecting appropriate Focus Areas for appraisal or process improvement.

Not all processes look alike or can be measured by the same standards or metrics. An organization may find it useful to include current "delays" and "queues" in the baseline process. During subsequent process improvement efforts, these allow a good starting point for cycle-time reduction.

If an organization defines its process from the point of view of the reponsibilities of its systems engineers, it should also define interfaces with the implementing disciplines of software engineering, electrical engineering, mechanical engineering, and more.

The model definition and usage is applicable to both programs and projects. The term "program" is equivalent to "project". "Program" refers to an inter-related set of activities to produce a product or highly related set of products. "Program" does not refer to an organizational infrastructure.

4.2 Model Architecture

The architecture of the SECM defines its structure and components. The architecture provides a context for understanding the details contained in the model, including the questions asked in the appraisal method, and the levels of capability defined in the model.

4.2.1 Model Structure and Components

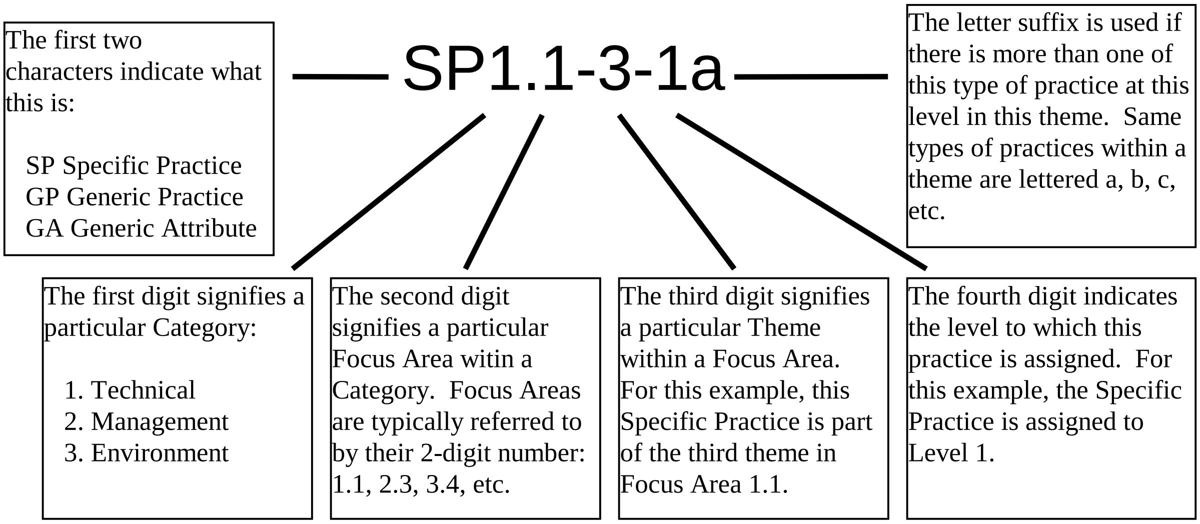
The SECM spans the breadth of Systems Engineering by defining Focus Areas (FA), each of which is defined by a set of unique Specific Practices (SP). Generic Characteristics (GC), which serve as differentiators of the maturity of an organization's capability to perform Systems Engineering, provide depth to the architecture and are grouped into levels of maturity.

The components of the model are defined as follows:

Categories	A Category is a natural grouping of Focus Areas. The three categories are: Technical, Management, and Environment.
Focus Areas	A Focus Area is a set of related, unique practices that address some aspect of Systems Engineering. There are 19 Focus Areas assigned to 3 Categories.
Themes	A Theme is a subdivision of a Focus Area that defines a related set of Specific Practices. Themes put lists of practices in context.
Specific Practices	A Specific Practice is an activity that is essential to accomplishing the purpose of a Focus Area or that helps accomplish the purpose of the Focus Area more effectively or efficiently. A Specific Practice will not be generic to all Focus Areas. Specific Practices are associated with specific levels of capability within each Focus Area.
Generic Practices	A Generic Practice is an activity that, when applied to the Specific Practices of a Focus Area, enhances the capability to perform those practices. Generic Practices are applicable to any Focus Area.
Generic Attributes	A Generic Attribute is an assessment of the effectiveness of the applied process and of the value of the products of the process. Generic Attributes are applicable to any Focus Area.

The model architecture is depicted in Figures 1, 2 and 3. Figure 1 illustrates the numbering scheme for the elements of the model.

Figure 1. Model Architecture: Numbering Scheme and Description



4.2.1.1 Categories of Focus Areas

Focus Areas fall into three natural groupings, or categories: Technical, Management, and Environment.

Technical Focus Areas

These practices are indicative of the technical aspects of the field of Systems Engineering. They correspond strongly with the definitions and practices contained in two prominent standards for Systems Engineering, EIA 632, *Processes for Engineering a System*, and IEEE STD 1220-1994, *IEEE Trial-Use Standard for Application and Management of the Systems Engineering Process*.

Management Focus Areas

These practices support the Technical Focus Areas through planning, control and information management. The practices of the Management Focus Areas couple corresponding practices from the standards for Systems Engineering with industry-wide best practices intended to promote efficiency, and hence cost-effectiveness, in the execution of the Systems Engineering process.

Environment Focus Areas

These practices enable sustained execution of the Systems Engineering process throughout an organization or enterprise and ensure the alignment of process and technology development with business objectives. These practices support the Technology and Management Focus Areas.

Figure 2. Model Architecture: Relationship of Categories, Focus Areas, Themes, Practices, and Attributes

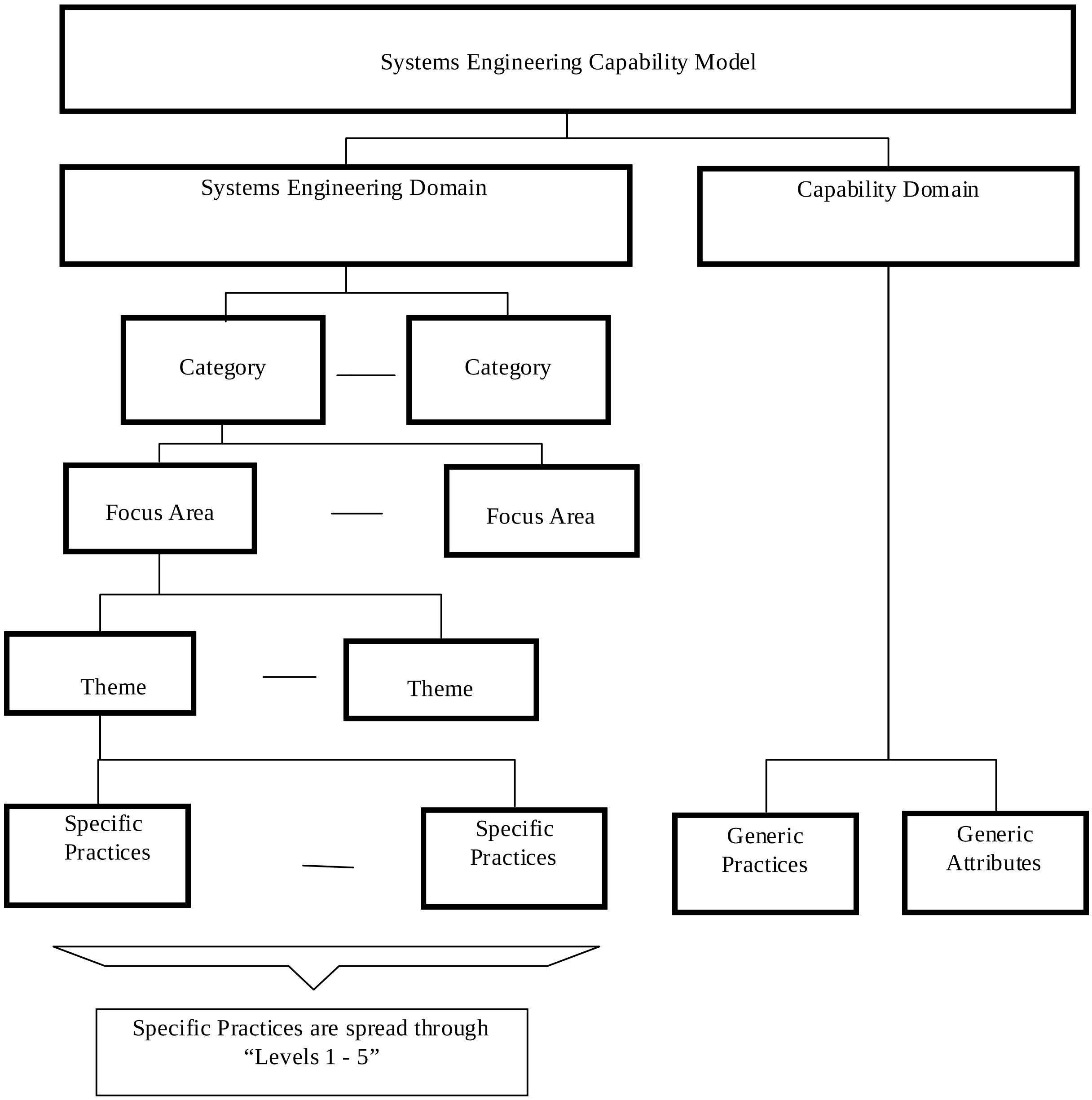
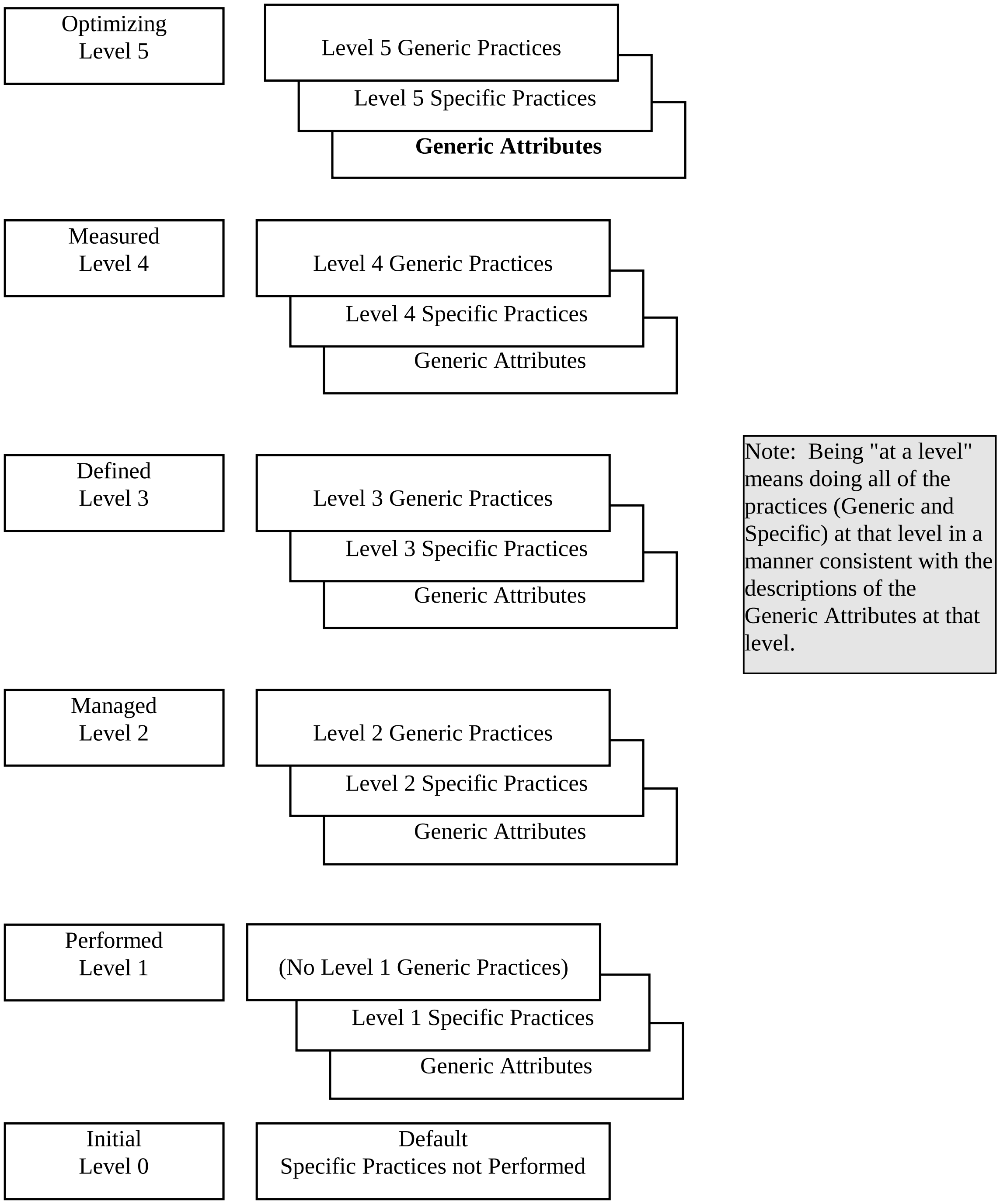


Figure 3. Relationship of Specific Practices and Generic Practices and Attributes to Levels of Capability



4.2.2 Capability Levels and Their Basis

There are six levels of capability. As shown in Table 1, each level of capability has practices and attributes associated with process and non-process characteristics, respectively. The lowest level is “Level 0”, and the highest is “Level 5”. Characteristics of Level 5 represent the highest, or optimal, level of performance.

Process attributes address the manner in which the practices of each Focus Area are performed. Non-process attributes are indicative of the judgement used in executing the Focus Area practices, the effectiveness of the work, and the value of the work and work products.

Generic Practices are somewhat related to the Software Process Improvement Capability dEtermination (SPICE) Project. Generic Attributes are derived from SECAM.

Capability levels are defined based on widely observed plateaus of performance that organizations typically achieve as they strive to improve their business processes and systems engineering activities. They can be used as goals, guidance, benchmarks, or other means to assist an organization in structuring its improvement efforts.

The combination of performance in conducting the Specific Practices, Generic Practices, and Generic Attributes of the Focus Areas in a category indicates the capability level of an organization. Figure 4 shows an example appraisal chart for the Technical Category.

Figure 4. Example Appraisal Chart for the Technical Category

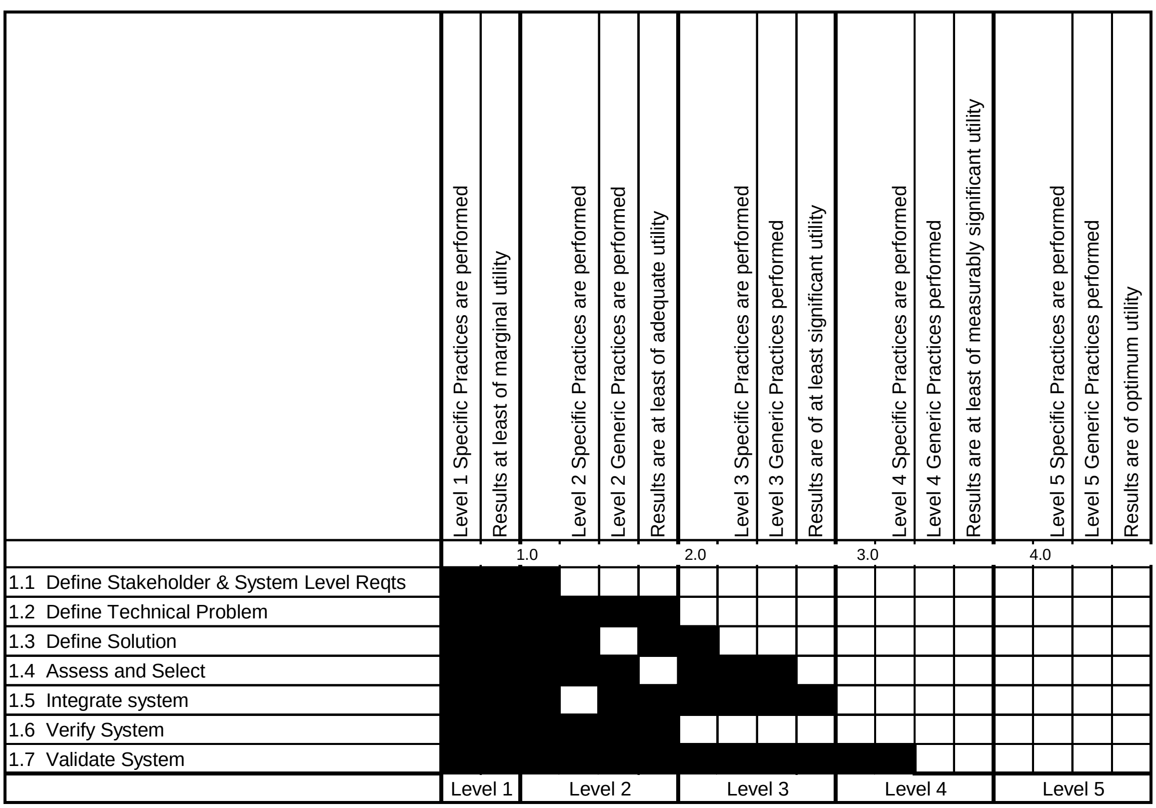


Table 1 - Capability Levels

Capability		Process	Non-Process
0	**Initial**	• Practices are not performed • general failure to perform activities • no easily identifiable work products • no proof tasks are accomplished	• activities and work products have little effectiveness or value • no assurance of success • information is difficult to identify • driving force for activities is indeterminate • no assurance of complexity management • no focus on the principles of systems engineering
1	**Performed**	• Specific Practices are performed • activities are done informally • non-rigorous plans and tracking • dependency on “heroes” • work products are in evidence • general recognition of need for activity	• activities are marginally effective and work products are of marginal utility • information is ad hoc • activities are driven only by immediate contractual or customer requirements • SE focus limited
2	**Managed**	• Specific Practices are performed, and performance is characterized by the Level 2 Generic Practices • policies define need for activities • processes are program specific • activities are planned, tracked, measured and verified • corrective actions are taken to assure the program specific process is followed • work products are reviewed for adequacy • defects are removed from work products • work products are controlled	• activities are adequately effective and work products are of adequate utility • key information managed • activities driven by customer and stakeholder needs in a suitable manner • SE focus is requirements through design
3	**Defined**	• Specific Practices are performed, and performance is characterized by the Level 3 Generic Practices • processes are well defined • the organization has a standard systems engineering process • tailoring guidelines exist for the standard systems engineering process • the standard systems engineering process is tailored and used by each program • tailoring is reviewed and approved • customer feedback is obtained	• activities are significantly effective and work products are of significant utility • consistent program success • information is managed and integrated • activities driven by benefit to program • SE focus is requirements through operation

Table 1 - Capability Levels (continued)

Capability		Process	Non-Process
3	**Defined (continued)**	• data are collected on the performance of the tailored process • qualitative process improvement is performed on both standard and tailored processes	
4	**Measured**	• Specific Practices are performed, and performance is characterized by the Level 4 Generic Practices • metrics are derived from data on the tailored process • the tailored process is quantitatively understood • performance of the tailored process can be predicted • tailored process induced defects are identified • measurable quality goals are established for systems engineering work products • causal analyses are performed for the tailored process • tailored processes are quantitatively improved • standard process continues to be qualitatively improved	• activities are measurably effective and work products are of measurably significant utility • all information fully integrated • activities driven by systems engineering benefit • SE focus on all phases of product life cycle
5	**Optimizing**	• Specific Practices are performed, and performance is characterized by the Level 5 Generic Practices • process effectiveness goals are established for the program based upon business objectives • causal analyses are performed for the standard process • standard processes are quantitatively improved • improvements to the standard process are flowed down into each tailored process	• activities are effectively balanced and work products effectively provide their intended utility • activities driven by systems engineering and organizational benefit • complexity management is fully scaleable • SE focus is product life cycle & strategic applications

4.3 Generic Characteristics

This section contains the Generic Characteristics (Generic Practices and Generic Attributes) that express the process and non-process capability aspects of the model. These Generic Characteristics are used to define the process capability levels as defined in the system architecture.

Process characteristics tend to describe, with increasing levels of capability, how well the process for a particular FA is defined, institutionalized, and followed. The process aspects of the capability levels of this model are derived from the International Standards Organization (ISO) Software Process Improvement Capability dEtermination (SPICE) Basic Practices Guide (BPG). The generic process characteristics of the SECM are aligned with the ISO SPICE BPG generic practices.

The non-process attributes tend to describe, with increasing levels of capability, how effective a process is and how valuable are its products. Non-process attributes are intended to provide a “sanity check” on the level of capability indicated by the process attributes.

The SECM defines a level of capability for performing the practices of a Focus Area by combining Generic Characteristics with Specific Practices. Generic Characteristics are comprised of process-oriented Generic Practices (GP) and non-process-oriented Generic Attributes (GA). Generic Characteristics are so-called because each one applies equally well to the practices of every Focus Area. For example, a Generic Practice that requires tailoring of a standard process is used for assessing Define Solution, Ensure Quality, Manage Technology, and so on. Figure 3 shows how Generic Characteristics combine with Specific Practices to define a capability level.

4.3.1 Generic Practices

A Generic Practice is an activity that, when applied to the Specific Practices of a Focus Area, enhances the capability to perform those practices. Generic Practices are applicable to any Focus Area, and are grouped into levels.

Level I. Performed

Description:

There are no Generic Practices in Level 1.

Level II. Managed

Description:

Activities are planned and tracked. Performance according to specified procedures is verified. Work products conform to specified standards and requirements. The organization uses measurement to track Focus Area performance, thus enabling the organization to manage its activities based on actual performance. The primary distinction from the Performed Level is that the performance of the process is planned and tracked. There are 2 Generic Practices at Level II.

GP 2.1 Follow recorded and approved plans and processes, that were developed to meet program performance goals, in implementing the Focus Area.

Description:

The following activities are examples of evidence of this generic practice:

- Plan the performance of the process in accordance with the established program goals (such as profit, customer satisfaction, schedule delivery and quality goals).
- Document the approach to performing the activities of the Focus Area that meets the program performance goals (such as profit, customer satisfaction, schedule delivery & quality goals) in standards or procedures.
- Use the documented plans, standards, or procedures in implementing the process for the program.
- Assign responsibilities for developing the work products and providing the services of the Focus Area.
- Allocate adequate resources including people, training, tools, budget, and time for performing the Focus Area.

Notes:

1. At this level reasonable processes are documented at the program level. A reasonable process is practiced, documented, enforced, trained, and measured.

2. Plans for Focus Areas in the Systems Engineering Technical and Management Categories may be in the form of a program plan, whereas plans for the Systems Engineering Environment Category may be in the form of an organizational policy.

3. Processes in an organization or on a program need not correspond one-to-one with the Focus Areas in the SECM. Therefore, a program's process addressing a Focus Area may be described in more than one way (e.g., policies, standards, and procedures). Similarly, a program's process description may span more than one Focus Area. Process measures should be defined in the standards, procedures, and plans.

Relationship to other Generic Practices:

A vertical theme occurs where a practice at a lower level relates to one at a higher level. Vertical themes occur in Generic Practices as well as in the Focus Area. In this case, GP 2.1 evolves to GP 3.3.

GP 2.2 Verify compliance with approved plans and processes, and take appropriate action when performance deviates from plan or when processes are not followed.

Description:

The following activities are examples of evidence of this Generic Practices:

- Verify compliance of the process and work products with applicable standards and procedures.
- Track the status of the Focus Area against the plan using measurement.
- Take corrective action as appropriate when progress varies significantly from that planned.

Notes:

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

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

Relationship to other Generic Practices:

The applicable standards and procedures were documented in GP 2.1.

This Generic Practice evolves to GP 3.4, GP 4.1, GP 4.2, GP 5.2, GP 5.3 and GP 5.4.

Level III. Defined

Description:

Activities are performed according to a well-defined process using approved, tailored versions of standard, documented processes. The primary distinction from the Managed Level is that the process is planned and managed using an organization-wide standard process. The organization's standard process is improved qualitatively. There are four Generic Practices at Level III.

GP 3.1 Standardize and record a well-defined FA process for the organization that is designed to meet specific business goals, and is based on experiences captured from previous programs.

Description:

Document a well-defined standard process or family of processes for the organization that describes how to implement the specific practices of the process. A well-defined process is characterized by:

- entrance criteria,
- inputs,
- standards and procedures,
- verification mechanisms (such as defect reviews),
- outputs,
- completion criteria, and
- metrics.

An organization's standard process (or processes) should be well defined. When the organization's standard process is well defined, a program may create a well-defined process for its use by tailoring the organization's well-defined process to meet program needs.

Notes:

1. The critical distinction between generic characteristics GP 2.1 and GP 3.1 in the Level 2 and Level 3 process descriptions is the scope of application of the policies, standards, and procedures. In GP 2.1, the standards and procedures may be in use in only a specific instance of the process, e.g., on a particular program. In GP 3.1, however, policies, standards, and procedures are being established at an organizational level for common use throughout the organization, and are termed the "standard process."

2. The organization may define more than one standard process to address differences among application domains, customer constraints, etc. These related standard processes are termed a "standard process family." The members of a standard process family are typically similar in their descriptions of how and in what order tasks are done. They typically differ in customer constraints, application domain (technology), etc.

3. Processes in an organization or on a program need not correspond one-to-one with the Focus Areas in the SECM. Therefore, a program's process addressing a Focus Area may be described in more than one way (e.g., policies, standards, and procedures). Similarly, a program's process description may span more than one Focus Area. Process measures should be defined in the standards, procedures, and plans.

Relationship to other Generic Practices:

The Level 3 standard process description is tailored in GP 3.2.

GP 3.2 Tailor the organization's standard process using standard guidelines to meet specific program or organizational needs.

Description:

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

Relationship to other Generic Practices:

The organization's standard process (family) is documented in GP 3.1. The tailored process definition is used in GP 3.3.

GP 3.3 Implement and improve the FA activities (i.e., tailored process) per established and approved formal procedures.

Description:

Use the tailored, well-defined process in implementing the Focus Area.

Relationship to other Generic Practices:

This GP is related to GP 3.2.

GP 3.4 Improve the organization's standard process using information from work product reviews and process compliance reviews.

Description:

Review the work products and the process implementation for compliance with requirements. Use data from reviews to improve the standard process.

Notes:

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

2. Defect reviews are sometimes called peer reviews. The purpose of defect reviews is to identify sources of error in early or interim systems engineering work products.

3. Data collection, analysis, and reporting are planned, and benefit both control and improvement activities. Data are used, as in GP 2.2, to track and initiate corrective action when deviations from planned performance are significant. Data are also used as a basis for identifying and prioritizing improvement opportunities. In addition, data should be collected on experiments or pilots of new or improved process elements, so as to understand their success or failure.

Relationship to other Generic Practices:

This practice is related to GP 2.2. GP 3.4 improves the standard process of GP 3.1. In addition, all of Level 4 Generic Practices builds on this Generic Practice.

Level IV. Measured

Description:

Level IV activities apply measurements to the process. Metrics are defined for the organization and programs, and mechanisms are in place to track program and organizational performance quantitatively, and to take corrective actions on the basis of those measures. There are two Generic Practices at Level IV.

GP 4.1 Collect and analyze metrics to determine the performance of the tailored FA activities.

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 Defined Level is that the tailored process is quantitatively understood and controlled.

Process performance of the tailored FA activities can be quantitatively characterized by (1) the extent to which the process is followed and (2) the cost, schedule, and quality of the product resulting from performing the process. To accomplish these performance characterizations for those repeatable processes under similar conditions: (1) collect data for each execution of the FA process and (2) analyze the aggregate data obtained from repeated execution of the FA process over similar programs to determine the process performance.

Notes:

1. This is a quantitative process capability based on a well-defined GP 3.1 and measured process. Measurements are inherent in the process definition and are collected as the process is being performed.

2. Methods and techniques for analyzing data and determining performance quantitatively may include the use of histograms, cause and effect diagrams, check sheets, Pareto diagrams, graphs, control charts, scatter diagrams, trend charts, or other statistical process control (SPC) improvement techniques.

GP 4.2 Take appropriate action to align tailored FA performance and expectations.

Description:

Take corrective action as appropriate when the process is not performing either as planned or as predicted.

Notes:

1. Actions may include cost analysis, specification changes, schedule or contract modifications, acceptance of risk, or implementation of process changes based on the analysis of the metrics.
2. For those processes that are repeatable, 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.

Level V. Optimizing

Description:

The organization establishes quantitative performance goals (targets) for process effectiveness and efficiency based on its business goals. The organization is able to continuously improve its process by gathering quantitative data from performing the processes and from piloting innovative ideas and technologies. The primary distinction from the Measured Level is that both the tailored process and the standard process undergo continuous refinement and improvement, based on a quantitative understanding of the impact of changes to these processes. There are four Generic Practices at Level V.

GP 5.1 Identify FA activities for which it is appropriate, and inappropriate, to quantify process repeatability.

Description:

A criteria for quantifying process repeatability is that a statistically significant number of cycles of the process can be performed under stable conditions. This generic practice requires that statistical or quantitative techniques be employed in a sound manner.

GP 5.2 Establish quantitative goals for improving the effectiveness of the standard process.

Description:

Establish quantitative goals for improving process effectiveness of the standard process family based on the business goals of the organization.

GP 5.3 Improve the organization's standard process based on data and metrics collected from a continuing program of process compliance reviews and work product reviews.

Description:

The goal of improving the standard process is to reduce common causes of variation. Continuously and quantitatively improve the process by changing the organization's standard process family to increase its effectiveness.

Notes:

The information learned from managing individual programs is communicated back to the organization for analysis. Changes to the organization's standard process family may come from innovations in technology or incremental improvements. Innovative improvements are often externally driven by new technologies. Incremental improvements will usually be internally driven and are often the results of tailoring the standard process by a program.

Relationship to other Generic Practices:

Improvements will typically be driven by the goals in GP 5.2

GP 5.4 Perform causal analysis of process and work product defects and eliminate causes of variation in quality, cost, and cycle time by changing the standard process.

Description:

Perform causal analysis of defects. Eliminate the causes of defects in the defined process selectively.

Notes:

1. Those who perform the process are typically participants in this analysis. This is a proactive causal analysis activity as opposed to reactive. Defects from prior programs of similar attributes can be used to target improvement areas for the new effort.

2. Defect causes are selectively eliminated because it may be impractical to perform causal analysis on all defects. In this case, some screening may be used.

Relationship to other Generic Practices:

GP 5.3 may be one source of improvements. Goals were established in GP 5.2.

4.3.2 Generic Attributes

Description:

The Generic Attributes address the effectiveness of the process and the value of the products of the process. Both of these are viewed over a range that is related to capability levels 1 - 5. With each increasing level, the worth of the product and process should increase. To be of maximum benefit, Focus Area (FA) activities must be performed in an effective manner, and the work products must be of value to the program. Good engineering judgment must be applied. The effectiveness of the activities and the value of the work products (marginal, adequate, significant, measurably significant, or optimal) are indicative of the level of performance.

GA-1*: Implement an effective set of FA activities (See Table 2).

GA-2*: Produce a valuable set of FA products (See Table 2).

*** The numbering of the Generic Attributes does not include a level. The levels are determined by the definitions in Table 2.**

Table 2 - Definition of Generic Attributes

Capability Level	Criteria	Generic Attribute 1: Effectiveness	Generic Attribute 2: Value
		A measure of the performance of an activity. SECM characterizes effectiveness as marginal, adequate, significant, measurably significant, and optimal. These are defined as:	*A measure of the desirability of the products of an activity. SECM characterizes value as marginal, adequate, significant, measurably significant, and optimal*. These are defined as:
1 Performed	Marginal	Effort is being expended but it is not clear that the benefits received for the effort invested are worth the cost of the effort. The effort could be removed without causing significant impact to the program or organization.	Products are generated by the activity, but it is not clear that the products are of use to those for whom they are intended. The products could be removed without causing significant impact to the program or organization.
2 Managed	Adequate	Effort is being expended and the activities provide reasonable benefit to the program or organization.	Products generated by the activity provide reasonable benefit to those that use them. Products providing adequate value are generally used by those for whom they are intended.
3 Defined	Significant	Effort being expended is obviously beneficial to the program or organization.	Products generated by the activity are obviously beneficial to those that use them. Products of significant value are avidly sought out and used by those for whom they are intended.
4 Measured	Measurably Significant	Effort being expended and the benefit are measured and found to be significant to the program or organization	The benefits of each product generated by the activity are measured and found to be of significant value to the program or organization.
5 Optimizing	Optimal	Effort expended is providing maximum benefit for the amount of effort, i.e., more effort results in a diminishing return to the program or organization.	Value of the products generated by the activity are of maximum utility to the program or organization.

5 Focus Areas

This section defines each of the nineteen Focus Areas, grouped into three categories, which comprise the EIA SECM. Each FA segment consists of a general description of the FA. The practices within each FA are organized into five levels of systems engineering capability; a sixth, and lowest, level of capability is a default level that contains no questions.

The numbering of each FA indicates the Category to which it belongs, as follows:

TC 1.0 Systems Engineering Technical Category

FA 1.1 Define Stakeholder and System Level Requirements
FA 1.2 Define Technical Problem
FA 1.3 Define Solution
FA 1.4 Assess and Select
FA 1.5 Integrate System
FA 1.6 Verify System
FA 1.7 Validate System

TC 2.0 Systems Engineering Management Category

FA 2.1 Plan and Organize
FA 2.2 Monitor and Control
FA 2.3 Integrate Disciplines
FA 2.4 Coordinate with Suppliers
FA 2.5 Manage Risk
FA 2.6 Manage Data
FA 2.7 Manage Configurations
FA 2.8 Ensure Quality

TC 3.0 Systems Engineering Environment Category

FA 3.1 Define and Improve the Systems Engineering Process
FA 3.2 Manage Competency
FA 3.3 Manage Technology
FA 3.4 Manage Systems Engineering Support Environment

FA 1.1 Define Stakeholder and System Level Requirements

*A Focus Area for the Systems Engineering **Technical** Category*

Description:

Define Stakeholder and System Level Requirements produces a validated set of system-level requirements (technical, support, interface) which reflect stakeholder needs and expectations. To accomplish this purpose, the Focus Area includes collection and coordination of stakeholder needs, development of an operational concept, and analysis of needs and requirements. These practices are supported by definition of alternative and preferred system concepts developed in FA 1.3 - Define Solution.

Stakeholders include customer/users, developers, producers, testers, suppliers, marketers, maintainers, disposers and others who may be affected by, or may affect, the system or product. Needs and expectations are elicited, stimulated, analyzed, and communicated to obtain the system technical requirements which constitute a common, quantitative, understanding of what will satisfy all stakeholders. Define Stakeholder and System Level Requirements includes engaging all stakeholders in an ongoing dialogue to translate needs and expectations into a concept of operations, system concept and a verifiable set of requirements, technological limitations, cost drivers, time drivers and risks, which the stakeholders understand and which provide the basis for agreements between the acquirer/customer/user and the developer. Suppliers and developers engage in a continuing solicitation of feedback from customer/users to establish and maintain the validity of requirements. Stakeholder needs, objectives, and requirements are analyzed in relation to mission operations, operational environments and desired system characteristics to derive alternative system concepts which unify the system feature, function, performance, and costs. System concepts are used to define and communicate requirements with the customer. The evolution and impact of evolving stakeholder requirements and technological constraints and features on systems/products are planned and managed.

Notes:

Since this Focus Area supports the dialogue between those performing systems engineering and the customer/user, all other Focus Areas will use this Focus Area to keep stakeholders informed throughout the program life cycle. Customer, as used here, denotes either a directly contracted customer, an internal customer, or a customer surrogate who represents a particular market segment in a market-driven, multi-customer industry. In addition to being identified and collected, needs and expectations are elicited (stimulated) from stakeholders by such means as demonstrating new technology or improved methods for solving operational problems. This FA has strong interaction with development of system concepts in FA 1.3 and integrated with all technical category FA's to provide the customer interface. Outputs of this FA, such as system requirements, are baselined and controlled by the practices of FA 2.7 - Manage Configurations.

References:

Not Applicable.

Themes of Define Stakeholder and System Level Requirements:

1.1-1 Stakeholder Needs
1.1-2 System Level Requirements

Theme Descriptions, Typical Work Products, and Practices:

1.1-1 Stakeholder Needs

Description:

Frequently, stakeholder needs and expectations are poorly identified or conflicting. Stakeholder needs and expectations, as well as customer limitations, must be clearly identified and prioritized. An iterative process is used throughout the life of the program to accomplish this. In the case of non-negotiated situations, the surrogate for the user or customer stakeholder is frequently the customer relations or marketing part of the organization. Environmental, legal, and other constraints which may be external to the customer must also be applied when creating and resolving the set of system requirements.

Comments:

Examples of techniques to identify and elicit needs include:

- Technology demonstrations
- 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, specifications, etc.
- Observation of existing systems, environments, and workflow patterns

Typical Work Products:

- Technical performance measures
- Draft requirements
- Needs statement
- Operational concepts
- Operational requirements
- Results of requirement reviews
- Needs issues/conflicts
- Stakeholder baseline
- Quality Function Deployment Needs Matrix
- Quality Function Deployment Requirements Matrix

Specific Practices:

SP 1.1-1-1	Identify, collect and baseline stakeholder needs, expectations and constraints.
SP 1.1-1-2a	Elicit or stimulate stakeholder needs.
SP 1.1-1-2b	Prioritize stakeholder needs, expectations and constraints.
SP 1.1-1-3a	Review, coordinate, and deconflict stakeholder needs and constraints.
SP 1-1-1-3b	Inform stakeholders on a regular basis about the status and disposition of needs, expectations, or measures of effectiveness.

1.1-2 System Level Requirements

Description:

The needs of all stakeholders (customers, users, suppliers, builders, testers, etc.) are considered in determining a set of system or product requirements. Analyses are 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 objectives of the analyses are to determine system concepts that will satisfy stakeholder needs and expectations, and then translate these concepts into system / product requirements. In parallel with this activity, the parameters that will be used to evaluate the effectiveness of the system or product are determined based on customer input and the preliminary system concept. Once a complete set of stakeholder needs and expectations and a preliminary operational and system concept are available, they are translated into system requirements.

Comments:

A role of systems engineering is often to help the customer formulate complete concepts. The customer's needs and expectations should be probed to ensure that the developer adequately understands them and has prioritized them correctly. Expression of logistics, support, maintenance, and training considerations are ways to capture system needs for feedback to the customer. Cost, feasibility, and risk should be considered in determining which requirements will be pursued. Examples of formal methods used to analyze needs include:

- Quality Function Deployment (QFD)
- Trade studies
- Mathematical techniques (design of experiments, sensitivity analysis, timing, sizing, simulation)
- Prototypes

System requirements may be initially provided by the customer. In this case, systems engineering performs a "validation" of these requirements by finding the inconsistencies or deficiencies, challenging infeasible or poor requirements, and negotiating changes as necessary. In other cases, the system engineering effort creates the entire set of system requirements. System requirements may be documented using a customer-specified format or development organization format.

Typical work products:

- System requirement documents or specifications
- Operational concept
- System cost objectives
- System installation, operational, maintenance and support concepts
- Disposal concepts
- Technical parameters
- Market-segment description
- System requirements baseline
- Requirement verification traceability matrix

Specific Practices:

SP 1.1-2-1a	Analyze and quantify functionality required by users.
SP 1.1-2-1b	Transform customer/user requirements into a set of system level requirements.
SP 1.1-2-1c	Define a system requirements baseline.
SP 1.1-2-1d	Obtain an agreement between acquirer and developer that system level requirements reflect their needs and expectations.

SP 1.1-2-2a	Develop operational concepts and scenarios, which include functionality, performance, maintenance, support and disposal as appropriate.
SP 1.1-2-2b	Review adequacy of system requirements to meet stakeholder needs with key stakeholders.
SP 1.1-2-2c	Review operational concepts and scenarios to refine and discover requirements
SP 1.1-2-2d	Record system requirement decisions that have a significant effect on cost, technical or schedule performance, and the rationale for the decisions.
SP 1.1-2-2e	Define the environment the system will operate in, including boundaries and constraints.
SP 1.1-2-3a	Negotiate an agreement between stakeholders and developers that system level requirements represent an optimum balance of their needs and expectations.
SP 1.1-2-3b	Allow for expansion and growth in system requirements.
SP 1.1-2-3c	Analyze and quantify functionality indicated by stakeholder requirements.
SP 1.1-2-4	Perform analyses, simulations or prototypes to assure that system requirements will satisfy stakeholder needs and expectations.

FA 1.2 Define Technical Problem

*A Focus Area for the Systems Engineering **Technical** Category*

Description:

Define Technical Problem includes development and evolution of the functional architecture, requirements management, establishment of lower level requirement baselines and generation of requirement specifications. This includes the further refinement of system level requirements into a complete set of detailed technical requirements. Requirements are derived or decomposed and allocated to system functions, people, associated processes or services. A hierarchy of logical entities (functions and subfunctions or object classes and subclasses) is established through iteration with the evolving operational concept. Requirements are refined, derived and allocated to the logical entities. The derivation of system requirements includes identifying requirements that are necessary from the standpoint of practical, economical system design, manufacturing, and support, and consideration of existing systems from which the system might evolve. System design is based on this complete set of system requirements. These activities are conducted iteratively to ensure requirements are satisfied, and as feasible solution alternatives are defined and refined.

Define Technical Problem analyzes requirements in light of the concept of operations (including functionality, support, maintenance and disposal) and derives a more detailed and precise set of requirements. Derived requirements arise from constraints, consideration of issues implied but not explicitly stated by the customer or user, and factors introduced by the developer's unique business considerations, regulations and laws

The allocated requirements are the basis for the definition of the system solution. The traceability of requirements is captured and maintained throughout the allocation process. User requirements are re-examined with each successive, lower-level set of requirements and functional architecture refinement. Trade studies are used to select the preferred set of requirements and functional architecture from competing alternatives at each level of detail. This is an iterative process. It occurs recursively at successively lower (more detailed) layers of a system's architecture, until sufficient detail is available to enable detailed design, acquisition, and testing of the system to proceed.

Requirements evolve throughout the system life cycle. Design decisions, subsequent corrective actions, and feedback from system operations, support, and disposal are analyzed for their impact on derived and allocated requirements . The subsystem requirements baseline, sometimes referred to as an allocated baseline, is established and maintained as a result of these activities.

Communication and coordination with stakeholders gives them visibility into the evolution of requirements, the evolution of the system operational concept (including maintenance, support and disposal), and system, managing their expectations and assuring them that their requirements are being satisfied. Opportunities to trade off system features for reduced cost and development time are continually presented to the customer.

Notes:

The practices of this Focus Area are executed interactively and in coordination with the activities of FA 1.1 - Define Stakeholder and System Level Requirements and FA 1.3 - Define Solution. FA 1.1 generates a well-defined understanding of the needs of the user, customer, and others who hold a stake in the system design. In turn, the system design, i.e., the detailed description of the hardware, software, and processes that comprise the system, is driven by the system requirements. Outputs of this FA, such as system requirements, are baselined and controlled by the practices of FA 2.7 - Manage Configurations. Requirements are identified in FA 1.1, FA 1.2, and FA 1.3, baselined and documented in FA 1.2, and formally configuration managed in FA 2.7.

References:

Not Applicable.

Themes of Define Technical Problem:

1.2-1	Problem Refinement
1.2-2	Requirements Analysis
1.2-3	Requirements Quality
1.2-4	Requirements Evolution and Maintenance
1.2-5	Feedback and Verification

Theme Descriptions, Typical Work Products, and Practices:

1.2-1 Problem Refinement

Description:

Define Technical Problem analyzes user requirements in light of the concept of operations and derives a more detailed and precise set of requirements. Derived requirements arise from constraints, consideration of issues implied but not explicitly stated in the customer/user problem baseline, and factors introduced by the developer's unique business considerations. All specified usage modes for the system are considered, and a time line analysis is generated for time critical sequencing of functions.

Typical Work Products:

- Derived requirements
- Interface requirements
- Functional architecture
- Requirement status (e.g., traceability, acceptance, issues)
- Updated stakeholder requirements
- Updated concept of operations
- Updated maintenance concept
- Updated support concept
- Updated disposal concept

Specific Practices:

SP 1.2-1-1a	Develop a detailed operational concept of the interaction of the system, the user, and the environment, that satisfies the operational, support, maintenance, and disposal needs.
SP 1.2-1-1b	Derive, from the system and other (e.g., environmental) requirements, requirements that may be logically inferred and implied as essential to system effectiveness.
SP 1.2-1-1c	Identify key stakeholder requirements and constraints that have a strong influence on cost, schedule, functionality, risk, or performance.
SP 1.2-1-2a	Identify and manage non-technical requirements concurrently with operational, functional, support, maintenance and disposal requirements.
SP 1.2-1-2b	Balance system and development cost and complexity, schedule, performance, and capabilities of existing designs and products in all trade studies using established criteria.
SP 1.2-1-2c	Capture relationships between requirements for consideration during change management and requirements allocation.
SP 1.2-1-2d	Maintain this status of requirements.
SP 1.2-1-3	Use validated models, simulations, and prototyping to reduce cost and risk of system development.

1.2-2 Requirements Analysis

Description:

A functional architecture is established, and the system requirements are allocated to system functions, objects, people, and processes. The allocated requirements are the basis for the synthesis of the system solution (the system design). The traceability of requirements to functions, objects, etc., is captured and maintained. The user requirements are re-examined with each successive, lower-level set of requirements and functional architecture, and the preferred system concept is refined. Trade studies are used to select the preferred set of requirements and functional architecture from competing alternatives at each level of detail. Prior requirements analyses are reviewed and updated. Inputs, outputs, and functional interfaces are also defined.

Typical Work Products:

- Derived and allocated baseline
- Functional architecture (functions or objects, behavior, data flow, allocated requirements)
- Interface specifications
- Trade studies
- Requirements traceability matrix

Specific Practices:

SP 1.2-2-1a	Partition requirements into groups, based on established criteria (such as similar functionality, performance, or coupling) to facilitate and focus the requirements analysis.
SP 1.2-2-1b	Consider the sequencing of time-critical functions both initially and subsequently during system component development.
SP 1.2-2-1c	Identify interface requirements associated with things external to the system and internally between functional partitions or objects.
SP 1.2-2-1d	Establish a derived requirements baseline, including the allocation of requirements to subsystems and system components.
SP 1.2-2-1e	Allocate requirements to functional partitions, objects, people, or support elements to support synthesis of solutions.
SP 1.2-2-2a	Maintain requirements traceability to ensure that lower level (derived) requirements are necessary and sufficient to meet the objectives of higher level requirements, and are consistent with the product's functional architecture.
SP 1.2-2-2b	Conduct trade studies or decision analyses to select between competing alternatives in all phases of the requirements process, including initial architecture development and subsequently in allocating requirements to lower levels of functional and physical architectures.
SP 1.2-2-3	Capture rationale for system level requirements, derived requirements, allocations, and traceability.

1.2-3 Requirements Quality

Description:

Quality criteria for system requirements as well as the system itself should be determined. Requirements quality is measured in terms of attributes such as correctness, completeness, stability, lack of ambiguity, and verifiability. System or product quality is concerned with delivery of a system, free of defects, that fully satisfies the customer's needs, as stated in the requirements. Completeness of requirements requires description of multiple views of the system (physical, functional, interface, timing, etc.) and comprehension of practical design constraints.

Typical Work Products:

- Requirement quality criteria
- Requirements issues
- Records of reviews and corrective actions
- Documented verification criteria
- System quality criteria

Specific Practices:

SP 1.2-3-1a	Analyze requirements to ensure that they are complete, correct, realizable and verifiable.
SP 1.2-3-1b	Develop and document system and subsystem verification criteria concurrently with requirements development.
SP 1.2-3-2	Formally review or inspect requirements for quality attributes including stability, lack of ambiguity, and traceability to the customer baseline.

1.2-4 Requirements Evolution and Maintenance

Description:

Requirements evolve throughout the system life cycle. Design decisions, subsequent corrective actions, and feedback from system operations are analyzed for their impact on derived and allocated requirements.

Typical Work Products:

- Requirements status
- Requirements decision database
- Trouble reporting system reports
- Change control process
- Requirements management plan
- Requirements traceability verification
- Requirements database

Specific Practices:

SP 1.2-4-1	Document changes to requirements.
SP 1.2-4-2a	Establish a process for formally and proactively controlling and managing changes to requirements, considering impact prior to commitment to change, gaining stakeholder buy-in, and tracking and closing out the actions and results.
SP 1.2-4-2b	Evaluate the impact of requirement changes from the standpoint of all stakeholders.

1.2-5 Feedback and Verification

Description:

Communication and coordination with stakeholders gives them visibility into the evolution of requirements, the evolution of the system operational concept, and system design. This activity continually manages their expectations and assures them that their requirements are being satisfied.

Typical Work Products:

- Records of stakeholder interactions

Specific Practices:

SP 1.2-5-1	Formally review requirements with stakeholders

SP 1.2-5-2a	Involve stakeholders in the process of requirements development.
SP 1.2-5-2b	Baseline (describe, capture, and control) and communicate requirements and functional architectures to all stakeholders.
SP 1.2-5-2c	Capture records of communications with stakeholders relative to requirements trade studies and allocations.
SP 1.2-5-3	Periodically review requirements and their relationship with system functional and physical architectures.

FA 1.3 Define Solution

*A Focus Area for the Systems Engineering **Technical** Category*

Description:

Define Solution is the process of transforming system level and lower level technical requirements into documentation of components and interfaces, such that their implementation and integration would satisfy the requirements. It includes the identification of key design constraints and issues, determination of the physical components and interfaces, allocation of requirements to system components, and assuring that the specified system components would meet the requirements if built/purchased/reused and integrated. System requirements are augmented with design constraints imposed by the selected system components. A key concept of Define Solution is the identification of alternatives and selection of a balanced solution in terms of cost schedule and technical performance. The practices of Define Solution are repeated for successive levels of system decomposition to arrive at detailed specifications of design components for implementation by appropriate disciplines. They are first employed to develop a system concept in support of system level requirements. The practices Define Solution make extensive use of FA 1.4 - Assess and Select and are expected to be performed interactively with the practices of FA 1.1 - Define Stakeholder and System Level Requirements and FA 1.2 - Define Technical Requirements.

Notes:

Activities of FA 1.3 - Define Solution are based on the System Level Requirements Baseline, which results from the activities of the FA 1.1 - Define Stakeholder and System Level Requirements Focus Area. Practices for the selection of a preferred solution are addressed in the Assess and Select Focus Area. This Focus Area is concerned with the physical (as distinct from logical/functional) architecture and design.

References:

Not Applicable.

Themes of Define Solution:

1.3-1 Design Synthesis
1.3-2 Requirements Allocation
1.3-3 Design Assurance

Theme Descriptions, Typical Work Products, and Practices:

1.3-1 Design Synthesis

Description:

Alternatives and their relative merits are considered in advance of selecting a solution per the Assess and Select Focus Area. Key requirements, design issues and constraints are established for use in alternative solutions analysis. Architectural features are considered which provide a foundation for system/product improvement and evolution (e.g. the choice of a database with certain interface characteristics might be selected with the expectation that interfaces to the database would be stable over several generations of product evolution). Reuse of Commercial Off The Shelf (COTS) and Non-Developmental Item (NDI) components are considered relative to cost, performance and product maintenance. The rationale for the selected solution and its design features are captured. The rationale for solutions not accepted is also captured and maintained for future reference as the system evolves.

Components of solutions and their interfaces are identified, documented, communicated and maintained for each level of system physical decomposition. External and component interfaces are fully defined in terms of origination, destination, stimulus, data characteristics, and electrical and mechanical properties. The capability of the design solution to meet requirements (design verification) is established throughout design definition, in accordance with the Verify Solution Focus Area. This is done through reviews, analysis, simulation, modeling, prototyping, similarity with existing systems or combinations thereof. Solution risk areas are identified, assessed, and managed, including development items, commercial off-the-shelf (COTS) items, government off-the-shelf (GOTS) items, and non-developmental items (NDI).

Comments:

The terms solution and preferred solution refer to the identification of components and their interfaces at any level of design decomposition. A thorough, unbiased consideration of alternatives, as opposed to justification of a preconceived solution, should be emphasized.

Typical Work Products:

- System description (component identifications and specifications/schematics)
- System concept
- Trade studies
- Analysis reports (e.g., performance, cost)
- Decision trees
- Operational concepts (including maintenance, support and disposal) for alternatives
- Make/buy decisions
- System architecture/design documents
- Captured nomenclature and features of components
- Interface descriptions (specifications, schematics)
- Data dictionaries
- Captured rationale for selection of solution
- Design verification reports and analyses
- Design risk analyses, reports and plans
- Product baseline

Specific Practices:

SP 1.3-1-1a	Capture and maintain a description of solution component features and constraints.
SP 1.3-1-1b	Generate alternative system concepts physical architectures, and design solutions and select a solution in accordance with FA 1.4 - Assess and Select.
SP 1.3-1-1c	Identify interfaces between design components and their requirements for specification and management in accordance with the practices of FA 1.5 - Integrate System.
SP 1.3-1-2a	Identify architectural or design issues that must be resolved to support successful development of the system.
SP 1.3-1-2b	Evolve the operational concept to a level of detail appropriate to each level of physical decomposition and input to the practices of FA 1.2 - Define Technical Problem for maintenance.
SP 1.3-1-2c	Record and maintain the solution description and rationale in a way that is accessible to all stakeholders.
SP 1.3-1-2d	Assign responsibilities for establishing the system architecture and design, and for enforcing it during development.
SP 1.3-1-3a	Fully define interfaces in terms of origination, destination, stimulus and data characteristics for software, and electrical and mechanical characteristics for hardware.
SP 1.3-1-3b	Plan for evolutionary use of purchased or non-developmental (COTS, GOTS, and reuse) items.

SP 1.3-1-3c	Develop system design alternatives which consider cost drivers, technology limitations and risk.
SP 1.3-1-3d	Develop timeline scenarios for system operation and user interaction for each alternative system design.
SP 1.3-1-3e	Establish a mechanism for determining if the prototyping of system functions is an appropriate part of the design process.
SP 1.3-1-3f	Establish a mechanism to identify design issues which should be subjected to decision analysis or trade studies throughout system development.
SP 1.3-1-3g	Capture the rationale for key (i.e., significant effect on cost, schedule or technical performance) decisions taken or defined.
SP 1.3-1-4a	Establish a mechanism to identify components which should be designed for reuse.
SP 1.3-1-4b	Develop system design alternatives which consider life cycle cost, complexity, system expansion, and growth.
SP 1.3-1-4c	Consider the evolution of requirement drivers and technology in selecting a preferred solution.
SP 1.3-1-5	Identify key architectural features which guide future system/product versions and upgrades.

1.3-2 Requirements Allocation:

Description:

Requirements are analyzed, assigned, or allocated to solution components. Initially requirements may be assigned to components without resolution of shared or redundant functionality or performance. When the performance or functionality of a requirement spans multiple components, analysis is performed to determine an allocation of the requirement's function or performance appropriate to each component. Derived requirements are formulated to specify the allocation appropriate to each component. As derived and allocated requirements are identified, they are updated and managed in accordance with FA 1.2 - Define Technical Problem. Requirements are verified for correctness and completeness against established criteria and in the context of operational concept threads, for both input requirements and requirements allocated to solution components. As requirements are analyzed and allocated, appropriate requirements are added to address all phases of the product's life cycle (e.g., production, support, and disposal) along with any design constraints associated with selected solution components. Cost, performance and time aspects of production and support requirements are communicated to stakeholders for consideration relative to business objectives. Requirements which have a significant impact on cost, schedule, or performance are designated "key requirements" for separate monitoring and input to risk analysis, as appropriate. Requirements are managed in terms of their status and volatility. The status of requirements, including issues, acceptance, and traceability to higher level requirements is maintained and communicated to stakeholders on a regular basis. The status of requirements is monitored and changes (or potential changes) are assessed for cost, schedule, and performance impacts.

Comments:

Operational concepts and scenarios for each level of system physical decomposition are used to achieve a common understanding of the design by reviewing requirement allocations in the context of operational concept thread reviews.

Typical Work Products:

- Subsystem specifications
- Thread analysis
- New/added requirements
- Requirement issues
- Allocation of key requirements to alternative solution components
- Requirement assignments
- Requirement allocations (including maintenance, support, and disposal)
- Allocation report
- Requirement change requests
- Requirement change impact assessments

Specific Practices:

SP 1.3-2-1a	Identify the assignment or allocation of requirements to design components and interfaces for recording and maintenance in accordance with requirement management practices of FA 1.2 - Define Technical Problem.
SP 1.3-2-1b	Identify traceability of derived requirements to parent requirements for recording and maintenance in accordance with requirement management practices of FA 1.2 - Define Technical Problem.
SP 1.3-2-2a	Identify requirement performance and functional allocations to design components and interfaces for recording and maintenance in accordance with the requirement management practices of FA 1.2 - Define Technical Problem.
SP 1.3-2-2b	Allocate key requirements to alternative solution components.
SP 1.3-2-3a	Identify and allocate appropriate derived requirements that address the effectiveness and cost of life-cycle phases following development, such as production and operation, to the extent they are compatible with business objectives.
SP 1.3-2-3b	Identify key requirements and design issues for separate tracking per the requirement management practices of FA 1.2 - Define Technical Problem, and for consideration by the practices of FA 2.5 - Manage Risk.
SP 1.3-2-3c	Review derived and allocated requirements for completeness and correctness against established criteria and in the context of operational concept threads or scenarios in accordance with the practices of FA 1.6 - Verify System.
SP 1.3-2-3d	Identify evolving requirement issues and their impacts to ongoing programs as inputs to the requirement management practices of FA 1.2 - Define Technical Problem.
SP 1.3-2-3e	Identify design constraints as requirements for each level of design.
SP 1.3-2-3f	Capture the rationale for requirement allocation decisions.

1.3-3 Design Assurance

Description:

Throughout the design process, activities are conducted and evidence captured to ensure that the evolving design, if implemented and integrated, will meet the requirements. Design assurance activities include evaluation of prototypes, simulation, modeling, informal and formal reviews. Design assurance activities are performed in accordance with the practices of FA 1.6 - Verify System.

Typical Work Products:

- Design review results
- Requirement and design issues
- Simulations or modeling results
- Prototypes
- Inspection reports
- Analysis results
- Risk items
- Review minutes

Specific Practices:

SP 1.3-3-1	Conduct internal and formal design reviews to verify that the design meets requirements.
SP 1.3-3-2a	Identify design risk areas for input to the practices of FA 2.5 - Manage Risk, and resolve risk items through analysis, prototyping, modeling or simulation.
SP 1.3-3-2b	Review component requirements to assure that the components are necessary and sufficient for meeting higher level requirements.
SP 1.3-3-3	Verify that the implemented design will meet functional and performance requirements, through analysis, prototyping, modeling or simulation.

FA 1.4 Assess and Select

*A Focus Area for the Systems Engineering **Technical** Category*

Description:

Assess and Select involves making good decisions [1] by (1) identifying issues which require the application of decision theory techniques in order to accomplish timely technical decision-making, (2) selecting an appropriate decision-making technique for technical issues (3) involving the right mix of technical disciplines, and (4) following the decision-making process during the system life-cycle. Assess and Select activities are applicable to budgeting, source selection, test planning, logistics, and production. In production, Assess and Select supports make-or-buy decision, development of manufacturing processes, selection of plant locations, and other decisions.

Technical issues requiring a decision-making process may be identified during any phase of a program. The objective should be to identify impending technical issues as early as possible in the program life cycle in order to maximize the time available to deal with each issue. Candidate technical issues for this area can be discovered using Risk Management activities. Additionally, technical alternatives for which there is no single preferred choice are also candidates for the application of these practices. Many methods exist to perform comparative studies of technical alternatives. Example methods are classical Trade-Off analysis, analytical hierarchical process, and quality function deployment. The selection of the appropriate decision making method should match the type and scope of the technical issue being analyzed.

The trade studies or decision activities must clearly identify the objective and requirements of the analysis, alternatives to be traded, the selected decision making method, and selection criteria. The development activity determines the measures to be considered for optimization and the relative weighting of those measures. Performance of the trade study will provide a recommendation as to which alternate should be selected. The selected alternative will provide the best balance among technical cost, schedule, and risk, and other important factors.

Notes:

Assess and Select may be invoked from any of the other Focus Areas. Inputs to Assess and Select include the alternatives or issues from FA 1.1 - Define Stakeholder and System Level Requirements, FA 1.2 - Define Technical Requirements, or FA 1.3 - Define Solution processes. Requirement, functional architecture, and physical architecture alternatives can be narrowed to a preferred alternative through application of analysis. Assess and Select also provides a means of resolving requirement (or functional or physical solution) conflicts in which no single candidate alternative will satisfy the requirements and constraints. There is a reluctance to use trade studies when the criteria are not conducive to objective analysis. Parameters such as credibility, national defense and political saleability of various alternatives are not always quantifiable. Since trade studies or other decision methods may not be straight forward in these situations, the analysis must be replaced with the judgment of the decision making body by balancing all criteria.

References:

1. *Systems Engineering Management Guide*, Jan. 1990, Defense Systems Management College, pg. 8-1 to 8-16.

Themes of Assess and Select:

Themes Descriptions, Typical Work Products, and Practices:

1.4-1 Selection of Appropriate Decision-Making Techniques

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. Decision-making techniques, ranging from consensus-based decisions to the use of probabilistic models and decision theory should be considered and selected appropriately.

Comments:

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

- Prototyping
- Simulation
- Modeling
- Trade study
- Literature searches
- Exploitation of prior analyses
- Elicitation of expert judgment

An example activity:

- Perform a sensitivity analysis on candidate solutions to determine if small variations in parameters will affect the outcome.

Typical Work Products:

- Decision tree
- Ground rules for decision-making
- Analysis techniques used to determine solutions
- Simulations and models

Specific Practices:

SP 1.4-1-1 Use structured decision making techniques to resolve technical issues.
SP 1.4-1-2 Select appropriate decision-making technique and record rationale for choice.

1.4-2 Consideration of Alternatives

Description:

Sufficient candidate solutions may or may not be furnished with the need for analysis. As the analysis proceeds, other alternatives should be added to the list of potential candidate solutions. The generation and consideration of multiple alternatives early in a decision-making process increases the likelihood that an acceptable decision will be made, and that consequences of the decision will be understood.

Typical Work Products:

- Trade-study library or archive
- Decision trees

Specific Practices:

SP 1.4-2-1	Consider all alternatives presented when making a decision.
SP 1.4-2-2	Identify alternatives for consideration in addition to those supplied with the problem statement.

1.4-3 Evaluation Criteria

Description:

The criteria upon which decisions will be based should be documented. The level of detail of documentation should be commensurate with cost, schedule, performance, and risk impacts of the decision. Assumptions should also be captured and evaluated for reasonableness and validity. The evaluation criteria should also comprehend the sensitivity of the analysis results, i.e., the rate at which the results change as input factors change.

Typical Work Products:

- Evaluation criteria

Specific Practices:

SP 1.4-3-1	Use established, documented evaluation criteria.
SP 1.4-3-2a	Evaluate the reasonableness and validity of assumptions.
SP 1.4-3-2b	Consider sensitivity of analysis results when establishing evaluation criteria.
SP 1.4-3-3a	Include technology limitations, environmental impact, and risks in evaluation criteria.
SP 1.4-3-3b	Include total ownership and life-cycle costs in evaluation criteria.
SP 1.4-3-3c	Capture the rationale for the selection and rejection of evaluation criteria.

1.4-4 Selection and Communication of Alternatives

Description:

A preferred approach is selected from among the alternatives, based on the established evaluation criteria. Communication of the decision is important in order to allow others to take actions based on the decision. Additionally, communicating the basis for the decision enables people to understand why a decision was made and is especially important if the trade-offs in the decision appear to be unfavorable from other points of view.

Typical Work Products:

- Decision results
- Trade study results
- Rationale for preferred solution
- Description of the preferred solution

Specific Practices:

SP 1.4-4-1a	Select a balanced solution based on established criteria.
SP 1.4-4-1b	Involve affected parties in the selection of preferred alternatives.
SP 1.4-4-2	Capture and communicate decisions and their rationale to affected parties.
SP 1.4-4-3	Record alternative solutions and the rationale for rejection.

FA 1.5 Integrate System

*A Focus Area for the Systems Engineering **Technical** Category*

Description:

The purpose of Integrate System is to ensure that system solution elements function as a whole. This involves identifying, defining, and controlling interfaces, as well as checkout of functions that are implemented across multiple system elements. The activities associated with Integrate System should be started early and continue throughout the entire product life cycle.

System Integration is more than just a one-time assembly of the system elements at the conclusion of design and fabrication. System Integration should be conducted incrementally, using an iterative process of “build-test-build”. This process may begin with analysis and simulations (e.g., threads, rapid prototypes, mechanical prototypes) and steadily progresses through increasingly more realistic incremental functionality until the final system is achieved. In each successive “build”, prototypes are constructed, tested, improved, and reconstructed based upon knowledge gained in the “testing” process. A successful integration strategy should use combinations of these techniques in an incremental manner. There is a high probability that the end product, integrated in this manner, will pass system verification and validation.

The basis for effective system integration is an integration strategy. An integration strategy should be developed early in the program, concurrently with system development, plans, and specifications. For complex systems, the integration strategy should be incremental and address the iterative process of “build-test-build”.

System integration is accomplished in part by controlling interfaces among system elements. Interface requirements, specifications, and detailed descriptions should be captured in a common repository. This repository should be accessible by everyone involved in the development of system elements that use the interface. As such, the repository provides an interface communication and baseline control function. Interface requirements in the repository are used to appropriately constrain the design of related subsystem elements and to ensure their compatibility. The content and structure of the data captured in the repository may vary greatly depending upon the type of interface (e.g., physical, logical, control, timing, and power).

Notes:

The integration strategy developed by Integrate System must be used when developing candidate architectures in FA 1.3 - Define Solution. Candidate architectures must support the desired integration strategy. The ability of a candidate architecture to support the integration strategy is also a factor in FA 1.4 - Assess and Select activities. Verification of system functions that have been implemented across a number of system elements is addressed in FA 1.6 - Verify System.

References:

Not Applicable.

Themes of Integrate System:

1.5-1 Integration Strategy
1.5-2 Interface Coordination
1.5-3 Integration Preparation
1.5-4 System Element Integration

Theme Descriptions, Typical Work Products, and Practices:

1.5-1 Integration Strategy

Description:

Develop an integration strategy and supporting documentation that identify the optimal sequence for receipt, assembly, and activation of the various components that make up the system. An optimal sequence is one that, based on business and technical factors, minimizes the overall cost, schedule and technical difficulties. The optimal sequence is recorded in an integration strategy. Attributes of an optimal integration approach include incremental iteration and establishment of an integration strategy early in product development. Incremental integration strategies provide for early assembly and checkout of components which provide a problem-free foundation for incorporation of other components as they become available or for prototypes of high risk components. Development of an early integration strategy, concurrently with requirements and design, allows the planning for, and application of, resources to support the selected integration strategy. The optimal sequence of integration may be bottom up, top down, or a hybrid thereof. Once established, the integration strategy must be periodically reviewed to ensure that variations in design or production and delivery schedules have not had an adverse impact on the sequence nor compromised the factors on which earlier decisions were made.

Comments:

System Integration focuses on verification of interface requirements associated with system elements, as contrasted with Verify System, which focuses on verification of the component requirements.

Typical Work Products:

- Integration strategy document
- Assembly / checkout drawings
- Sequence and rationale for selected assembly
- Systems Integration and Test Plan

Specific Practices:

SP 1.5-1-1	Develop an integration strategy.
SP 1.5-1-2	Document the integration strategy as part of an integration plan.
SP 1.5-1-3a	Develop the integration plan early in the program.
SP 1.5-1-3b	When multiple teams are involved with system development, establish and follow a formal procedure for coordinating integration activities.
SP 1.5-1-4a	Review the integration strategy on a continuous basis.
SP 1.5-1-4b	Capture rationale for decisions taken and deferred.
SP 1.5-1-5	Improve standard integration strategies based upon rationale for decisions which resulted in improved integration performance.

1.5-2 Interface Coordination

Description:

The objective of interface coordination is to establish interface definitions between related system elements as early as possible. Many integration problems arise from unknown or uncontrolled aspects of interfaces. Coordination of interface requirements, specification, and design assures that implemented interfaces will work together. Therefore, system and subsystem interfaces should be specified as early as possible in the development life cycle. Interface specifications should address logical, physical, electrical, mechanical, human, and environmental parameters as appropriate. Intra-system interfaces should be the first design consideration for developers of the system building blocks. Interfaces may be derived from previous development efforts or developed in accordance with interface standards for the given discipline or technology. Novel interfaces should be constructed only for compelling reasons. Interface requirements for each building block should be verified against the interface specification. When exceptions are identified, further interface coordination may be required to resolve the interface conflict between system elements.

Comments:

Interface coordination may be accomplished through the use of an Interface Control Working Group (ICWG). The ICWG consists of representatives for each of the system components. Working together, these representatives are able to make decisions which optimize the system. The system architecture developed by the Define Solution Focus Area provides scope and context for this activity. Examples of interface specification data include data element descriptions, direction, and frequency. Definition of mechanical and environmental interface requirements is appropriate during the architecture definition phase, especially for interfaces to existing systems or subsystems.

Typical Work Products:

- Exception reports
- Exception resolution reports
- Interface specifications
- Interface control documents
- Interface requirements specifications

Specific Practices:

SP 1.5-2-1a	Coordinate interface definition, design, and changes between affected groups and individuals throughout the life cycle.
SP 1.5-2-1b	Identify interface requirements baselines.
SP 1.5-2-2a	Review interface data.
SP 1.5-2-2b	Ensure complete coverage of all interfaces.
SP 1.5-2-3a	Capture all interface designs in a common interface control format.
SP 1.5-2-3b	Capture interface design rationale.
SP 1.5-2-3c	Store interface data in a commonly accessible repository.
SP 1.5-2-4	Review the adequacy of interface documentation periodically.

1.5-3 Integration Preparation

Description:

Ensure that the assembly of the system elements into larger and more complex aggregate elements is conducted in accordance with the planned integration strategy. The timely receipt of needed system elements and the involvement of the right people contribute to the successful integration of the aggregates of system elements which comprise the system. Verification of the aggregates is explicitly addressed in FA 1.6 - Verify System.

Typical Work Products:

- Integration and Test Plan review summary
- Acceptance documents
- Delivery receipts
- Checked packing lists
- Exception reports

Specific Practices:

SP 1.5-3-1a	Verify the receipt of each system element (component) required to assemble the system in accordance with the physical architecture.
SP 1.5-3-1b	Verify that the system element interfaces comply with the interface documentation prior to assembly.
SP 1.5-3-2	Coordinate the receipt of system elements for system integration according to the planned integration strategy.

1.5-4 System Element Integration

Description:

Integration of system elements should proceed in accordance with the integration plan. Upon receipt, each system element should be verified to be compliant with its interface requirements. Aggregates of system elements are assembled into larger, more complex system elements. These aggregates are checked for correct inter-operation. This process continues until system integration is complete. If during this process exceptions are identified, the exception should be documented and a resolution process initiated.

Typical Work Products:

- Exception reports
- Interface checkout reports
- Integration summary reports

Specific Practices:

SP 1.5-4-1a	Assemble aggregates of system elements in accordance with the integration plan.
SP 1.5-4-1b	Checkout assembled aggregates of system elements.
SP 1.5-4-4	When multiple organizations are involved in system integration, periodically assess the quality of their mutual interaction to improve the program-level integration effort.

FA 1.6 Verify System

*A Focus Area for the Systems Engineering **Technical** Category*

Description:

Verify System ensures system stakeholders that the incremental and final work products and system conform to requirements. It encompasses themes of integrated verification planning, preparation for verification, incremental verification, and analysis and actions.

System verification is a stepwise approach to ensuring that each element of a system satisfies its requirements. Verification at each level of the system hierarchy substantially increases the likelihood that the complete, integrated system will satisfy the system-level requirements. Just as requirements are successively developed for each level of system decomposition (i.e., system, segment, subsystem, unit, components, etc.), the system verification process is successively applied to determine, for a given level of development activity, that the implementation satisfies the requirements specified at that level and that interfaces are as defined in the interface control documents. Verification is accomplished via a combination of methods: test, analysis, simulation, demonstration, and inspection.

System verification has several components, including functional, physical (including environmental), and interface verification, and work product verification. Functional system verification involves system performance testing, which verifies performance with respect to requirements, and qualification testing, which verifies system performance within its specified operational environment (e.g., temperature, vibration, electromagnetic interference). Physical system verification determines that the system, as it has been built, complies with a configured set of requirements, specifications, or drawings.

Notes:

Verification and validation activities are very similar, but they address different issues. Validation confirms that the system, as built (or as it will be built), will satisfy the user’s needs, whereas verification addresses whether the system, its elements, its interfaces, and incremental work products satisfy their requirements. In the Systems Engineering process, requirements become a surrogate for user, operator, maintainer, and disposer needs and include them with the concerns of all other system stakeholders. Verification ensures conformance to those requirements, and validation ensures the requirements and the system implementation provide the right solution to the customer’s problem. Each process occurs incrementally, beginning with validation of the understanding of the user’s needs and customer involvement in the system definition process.

References:

Not Applicable.

Themes of Verify Solution:

1.6-1 Integrated Verification Planning
1.6-2 Preparation for Verification
1.6-3 Incremental Verification
1.6-4 Analysis and Actions

Theme Descriptions, Typical Work Products, and Practices:

1.6-1 Integrated Verification Planning

Description:

The verification process begins concurrently with the development of system requirements and development plans. Verification planning also covers incremental and final Focus Area work products. Comprehensive verification planning is required to assure that all aspects of system testing and verification, including developmental testing, requirements quality verification, factory test, first article qualification, Built-In-Test (BIT), bench-testing, and operational or field testing are conducted. This up-front planning is necessary to ensure that verification provisions are embedded in system requirements, design and developmental plans, and schedules. Verification is an inherently incremental process since it occurs throughout a system's development, beginning with verification of the quality of the requirements and other work products, progressing through the verification of components and the integrated system, and ultimately to the operation of system in its intended environment.

Typical Work Products:

- System verification plan
- Schedules for in-progress and final reviews of work products
- Component verification plans
- COTS verification plans
- Verification scenarios

Specific Practices:

SP 1.6-1-1	Plan the set of comprehensive, integrated verification activities, addressing all certification requirements, objectives, resources, facilities, special equipment, and schedules applicable to the system development.
SP 1.6-1-2a	Include realistic operational and environmental scenarios in system verification plans.
SP 1.6-1-2b	Review verification plans early with peers within the developer's organization and with other system stakeholders to assess risky aspects of system development and to agree on alternative courses of action in the event of failures while conducting verification.
SP 1.6-1-3	Require development of verification plans in organizational policy.

1.6-2 Preparation for Verification

Description:

Preparation for verification begins with involvement in the requirements definition process for the purpose of ensuring that system requirements are verifiable. Verifiability is a measure of whether a requirement is verifiable at some level of system integration (system, subsystem, component, etc.). Preparation includes ensuring that an appropriate method of verification is assigned concurrently with the establishment of each requirement. Methods of verification include, but are not limited to, inspection, analysis, simulation, testing, and demonstration. Preparation also entails definition and acquisition of test equipment, software, and facilities, including definition of how both these test assets and the items to be tested will be used and reused throughout system development and integration.

Typical Work Products:

- Verification procedures
- Verification assignments and responsibilities
- Verification support equipment
- Verification evaluation criteria

- Test environment definition

Specific Practices:

SP 1.6-2-1	Define the methods, process, and evaluation criteria by which the systems, subsystems and work products are verified against their requirements in a written plan.
SP 1.6-2-2	Identify the individual or team responsible for verification in the verification plan and assign qualified personnel per the plan.
SP 1.6-2-3a	Adjust system requirements and development plans appropriately according to risks of failing system verification.
SP 1.6-2-3b	Acquire test equipment and software and items to be tested according to a comprehensive strategy that enables reuse.
SP 1.6-2-3c	Validate test or analysis procedures and support facilities.

1.6-3 Incremental Verification

Description:

Verifying developed products incrementally ensures that problems are found, and defects contained, as early in the development process as possible, saving the considerable cost of fault isolation and rework associated with trouble-shooting problems in a complex, integrated system. Verification may address identified work products, such as in-progress requirement, design, and component specifications; use of formal and informal reviews and audits; as well as the system and test equipment being developed. Incremental verification requires that the developer be capable of managing multiple configurations of systems, products, and documentation.

Typical Work Products:

- Results of work product verifications
- Results of subsystem and component verifications
- Results of system verification
- Demonstrations

Specific Practices:

SP 1.6-3-1a	Perform re-verification of corrected deficiencies and changed requirements and designs.
SP 1.6-3-1b	Inspect implemented, purchased, and reused components to verify they meet requirements.
SP 1.6-3-2a	Test new and unproven designs (i.e., highest risk) at the lowest assembly level to verify their compliance with established requirements early in the development life cycle.
SP 1.6-3-2b	Review the incremental verification results vis-à-vis requirements with key stakeholders on an on-going basis.
SP 1.6-3-2c	Verify system, subsystem, and work products against requirements established in an earlier phase.
SP 1.6-3-2d	Perform incremental verification on systems, subsystems, and work products.

1.6-4 Analysis and Actions

Description:

Verification activities are executed, and the resulting data are collected and analyzed against the defined verification criteria. Reports state verification results in terms of the system requirements, indicating whether or not requirements were met and, in case of deficiencies, assessing the degree of success or failure, categorizing causes of failure, and indicating what actions were taken as a result.

The collected test, inspection, or review results are compared with established evaluation criteria. Each failure is assessed, and a determination is made as to whether to proceed or to rework and retest. The verification environment is carefully controlled to provide for replication, analysis of results, and reverification of problem areas.

Typical Work Products:

- Test reports
- Analysis reports
- Corrective actions to work products
- Corrective actions to components and subsystems

Specific Practices:

SP 1.6-4-1	Compare the collected test, inspection, or review results with established evaluation criteria to assess the degree of success.
SP 1.6-4-2a	Involve all product stakeholders in the review of system verification results and issues.
SP 1.6-4-2b	Inform stakeholders of the results of verification activities.
SP 1.6-4-3	Use verification results to compare actual measurements and performance to technical performance parameters.

FA 1.7 Validate System

*A Focus Area for the Systems Engineering **Technical** Category*

Description:

System validation is an "end-to-end" approach to ensure that the completed, integrated system will operate as needed in the environment for which it is intended. It is a measure of customer satisfaction, given the customer's operational need. Absolute system validation can only be accomplished using the actual system in its intended environment. However, validation issues can be discovered early in the development life cycle through the use of early validation activities. Validation should be done in a realistic operational environment. Such an environment could include personnel, procedures, data packages, and logistical support. Validation issues typically include incomplete satisfaction of the mission requirements but may also consist of unanticipated or unintended functions or behaviors.

Validation activities use approaches similar to verification (i.e., test, analysis, simulation, etc.). Prototyping techniques (e.g., rapid or sluggish prototyping) may also be used. Prototypes may be physical models or mockups used to determine how a man-machine interface might be optimized, or simulations that provide insight into the operation or behavioral characteristics of a system. Validation and verification activities often run in parallel and may use much of the same test environment. The difference is that verification is demonstrating compliance with requirements and validation is demonstrating satisfactory compliance with the operational need. In other words, verification assures you built it right, while validation assures you built the right thing.

Notes:

1. System Validation primarily addresses the operational and system needs developed as part of FA 1.1 - Define Stakeholder and System Level Requirements.

2. Corrective actions resulting from validation activities are handled in FA 2.2 - Monitor and Control.

References:

Not Applicable.

Themes of Validate System:

1.7-1 Validation Strategy
1.7-2 Requirements Validation
1.7-3 Product Validation
1.7-4 Analysis and Actions

Theme Descriptions, Typical Work Products, and Practices:

1.7-1 Validation Strategy

Description:

The purpose of developing a validation strategy is to establish the environment, operational scenario, test procedures, inputs, outputs, expected results, and evaluation criteria for validation of the system. Development of a validation strategy involves the stakeholders in determining the approach, schedule, system configuration, environment, and resource requirements for operational evaluation of the system. The strategy should include verification, configuration control, and maintenance of the test equipment and environment. System validation takes into account the customer as user/operator of the system during testing. It includes both structured and unstructured operation of the system or product by the user and maintainer, and defines the type of data to be collected, analyzed, and reported.

Typical Work Products:

- Validation plan
- Operational, maintenance and support test and evaluation plan
- Test environment definition
- Simulation requirements
- Validation procedures

Specific Practices:

SP 1.7-1-1	Develop a strategy for system validation.
SP 1.7-1-2	Define requirements for a realistic operational, maintenance, and support environment.
SP 1.7-1-3	Formally document the environment, operational scenario, test procedures, inputs, outputs, expected results, and evaluation criteria for the system validation plan.

1.7-2 Requirements Validation

Description:

Validation activities start early in the program and are performed according to defined procedures. Validation is performed to ensure the stakeholder needs have been captured in the requirements. The system requirements are analyzed to ensure that the defined set of requirements is consistent with the baselined operational need. This process may be repeated incrementally at any stage of the design to ensure that the developing design is consistent with the intended mission(s). The results are captured to support analysis and comparison. The validation environment should be carefully controlled to provide for replication, analysis of results, and re-validation of problem areas.

Typical Work Products:

- Inspection results
- Simulation results
- System validation data
- Validation exception reports

Specific Practices:

SP 1.7-2-1	Conduct early requirements validation in some fashion on the program to reduce the risk of failing system validation.

SP 1.7-2-2	Provide appropriate tools to support system requirement validation activities (e.g., rapid prototyping, simulation, decision making, etc.).
SP 1.7-2-3	Factor system validation issues into risk analysis.
SP 1.7-2-4	Review the results of early validation periodically to assess the adequacy of the system design as it matures, with corrective action taken as necessary.

1.7-3 Product Validation

Description:

Validation is performed to ensure the baselined stakeholder needs have been realized in the work product or system. During product validation the goal is to demonstrate that the as-built product actually performs its intended function and to identify any unintended behaviors that may be detrimental. Product validation activities usually include structured scenario testing as well as 'break it' testing to uncover unintended behaviors or sneak circuits. Results are captured to support analysis and comparison with expected results. The validation environment should be carefully controlled to provide for replication, analysis of results, and re-validation of problem areas.

Typical Work Products:

- Test reports
- Validation cross-reference matrix
- Test results
- Demonstrations
- Analysis reports
- Validation exception reports
- Trouble reports

Specific Practices:

SP 1.7-3-1	Perform operational test and evaluation in some manner.
SP 1.7-3-2a	Perform operational, maintenance, and support test and evaluation.
SP 1.7-3-2b	Provide appropriate tools to support system validation activities, both simulations and actual systems.

1.7-4 Analysis and Actions

Description:

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 validation criteria. Analysis reports indicate whether or not requirements were met and, in the case of deficiencies, assess the degree of success or failure and categorize the probable cause of failure. The collected test, inspection, or review results are compared with established evaluation criteria, to determine whether to proceed or to rework and retest.

Typical Work Products:

- Test deficiency reports
- Test incident reports
- Design change requests
- Contract deviation request

Specific Practices:

SP 1.7-4-1	Assess system validation issues for their impact on the program.
SP 1.7-4-2a	Coordinate the resolution of validation issues among affected projects within the program.
SP 1.7-4-2b	Use the results of early validation to support tracking and oversight of technical performance parameters.
SP 1.7-4-3a	Include early validation activities as part of concept definition to reduce risk of specifying invalid requirements.
SP 1.7-4-3b	Include system validation issues (e.g., unanticipated or unintended functions or behavior) as an integral part of all formal, system level design reviews.

FA 2.1 Plan and Organize

*A Focus Area for the Systems Engineering **Management** Category*

Description:

Plan and Organize involves the identification of needs and constraints at the program level. The results of planning may be classified in terms of technical requirements and program requirements. These requirements define the technical and program structure required to bring a system into being. Planning includes: program requirements definition; identification, integration and scheduling of all engineering functions and tasks; work breakdown structure (WBS) development; organizational structure definition (as related to the program); and descriptions of, or references to, key policies and procedures. Planning is documented in a technical management plan which sometimes references other planning documents.

The technical management plan relates the technical requirements to program requirements, providing the structure to guide and control the integration of engineering activities needed to achieve the systems engineering objectives consistent with a top level management plan for the program. The technical management plan addresses planning for three basic areas:

- Technical program planning & control,
- Systems engineering process,
- Engineering specialty integration.

The technical management plan includes an event-driven plan and a calendar-based plan.

An event-driven plan is generated that lays out the core technical portion of the program, process descriptions, and significant events. The event-driven plan documents the significant accomplishments necessary to complete the program's efforts and ties each accomplishment to a key program event. This event driven plan is included (sometimes by reference) as part of the technical management plan. Each significant event may be thought of as a function with defined tasks to be accomplished. Entrance criteria are defined to start the event (function) and accomplishment (exit) criteria are established to determine the completion of the event (function).

A calendar-based plan is generated for significant events and activities within the program and is included (sometimes by reference) as part of the technical management plan. In non-complex programs, this calendar-based plan may be a Gantt chart only. In complex programs, the calendar-based plan may be both a Gantt chart and a network chart that relates dependencies among tasks and events and permits the determination of a critical path.

Organizing to execute the system development involves entirely defining organizational structure (teams, work groups, programs, etc.), establishing the responsibilities, authority and accountability of each, and clearly defining the interfaces among them. Organize includes organizational structure for multi-disciplinary teamwork. This should be an iterative process.

Notes:

FA 2.1 - Plan and Organize provides the planning and organization necessary to accomplish the technical portion of a program, and therefore relates to all Focus Areas in the Technical and Management categories. It relates to FA 3.1 - Define and Improve the Systems Engineering Process in that a standard systems engineering process needs to be tailored to meet specific program needs; the tailored process then is incorporated into the program planning.

References:

Not Applicable.

Themes of Plan and Organize:

2.1-1	Critical Resources
2.1-2	Technical Approach
2.1-3	Work Breakdown Structure, Estimation, and Task Description
2.1-4	Schedules
2.1-5	Technical Management Plan

Theme Descriptions, Typical Work Products, and Practices:

2.1-1 Critical Resources

Description:

The program's staff identifies the resources that are essential for its technical success for both the current and subsequent life cycle phases. Critical resources are identified, and contingency plans are made in the event that they are not available.

Typical Work Products:

- List of critical resources
- Contingency plans

Specific Practices:

SP 2.1-1-1	Identify resources that are critical to the technical success of the program.
SP 2.1-1-2a	Reconcile the level of technical work required for the program to the available level of funding or projected market potential.
SP 2.1-1-2b	Assign responsibility for developing the technical management plan.
SP 2.1-1-2c	Designate a systems engineering first-line manager or team leader to be responsible for negotiating technical commitments.

2.1-2 Technical Approach

Description:

A technical approach is selected for the effort to be accomplished. The approach selected should reflect the needs of the particular phase of the product life cycle that the program is currently in, as well as future life cycle phases.

Typical Work Products:

- Life cycle model
- Systems engineering process selected and tailored
- Technical approach

Specific Practices:

SP 2.1-2-1a	Determine a technical approach for the program.

SP 2.1-2-1b	Estimate the magnitude and technical feasibility of the program.
SP 2.1-2-2a	Identify technical activities for the entire life cycle of the program.
SP 2.1-2-2b	Identify key technical performance parameters.
SP 2.1-2-2c	Establish thresholds or profiles for key technical performance parameters.
SP 2.1-2-3	Identify and define a system life cycle with predefined stages of manageable size.

2.1-3 Work Breakdown Structure, Estimation, and Task Description

Description:

The program estimates the technical scope, size, and cost of the effort to be accomplished. Depending upon the life cycle phase, this may be accomplished using parametric cost models, analogy models, or a bottom-up estimate. The costs are organized into a work breakdown structure, which is a tree-like structure that permits summing of subordinate costs for tasks, materials, etc., into their successively higher level “parent” tasks, materials, etc. For each element of the work breakdown structure, a description of the task to be performed is generated.

Typical Work Products:

- Estimates for the size of the effort, e.g.,
  - Lines of code, function points, staff months
  - Number of electronics cards
- Labor costs by labor grade
- Rationale to support labor costs
- Work Breakdown Structure
- WBS dictionary
- Program organizational structure (team structure, staff assigned)

Specific Practices:

SP 2.1-3-1a	Generate a work breakdown structure for the program that defines logical units of work to be managed at the program level.
SP 2.1-3-1b	Develop cost estimates for the technical aspects of the program.
SP 2.1-3-1c	Generate documented and approved statements of work for systems engineering activities.
SP 2.1-3-2a	Define systems engineering work products, including data requirements, and activities in a traceable and accountable manner, including data requirements.
SP 2.1-3-2b	Ensure the technical management plan provides form and context for the planned technical activities and identify products.
SP 2.1-3-2c	Ensure the work breakdown structure covers all the tasks and products necessary for the program.
SP 2.1-3-3a	Derive estimates for the size and cost of the systems engineering work products and efforts based upon historical data.
SP 2.1-3-3b	Consider whether a system is precedented or unprecedented when generating estimates of the engineering effort.
SP 2.1-3-3c	Capture the basis or rationale for systems engineering planning and estimates.
SP 2.1-3-3d	Assure that the work breakdown structure reviewed is complete- and consistent with the system or product structure.

2.1-4 Schedules

Description:

The program develops one or more schedules that tie the technical activities to specific timelines. For a simple development effort, Gantt schedules (charts) may suffice. Complex development efforts usually use network schedules that reflect relationships between technical activities and identify a critical path for accomplishing the effort.

Typical Work Products:

- Gantt schedules (charts)
- Network schedules

Specific Practices:

SP 2.1-4-1	Develop schedules for the current life cycle phase as a part of the planning activities.
SP 2.1-4-2a	Develop top level schedules for the remaining life cycle phases of the program.
SP 2.1-4-2b	Address task dependencies as a part of scheduling.
SP 2.1-4-3	Provide traceability between the schedule (calendar-based plan) and the event-driven plan.

2.1-5 Technical Management Plan

Description:

A technical management plan is generated for the program that defines all aspects of the technical effort. The technical management plan ties together in a logical manner: life cycle considerations, application of the systems engineering process to accomplish the task, technical activities, entry and exit criteria for key or critical technical activities (integrated master plan (IMP)), and schedules for the technical activities and milestones.

Typical Work Products:

- Technical management plan (technical part of Program Management Plan
- Integrated Master Plan (IMP)
- Integrated Master Schedule (IMS)
- Gantt charts
- Network diagrams
- Systems Engineering Management Plan (SEMP)

Specific Practices:

SP 2.1-5-1a	Develop a technical management plan for the program.
SP 2.1-5-1b	Ensure there are clear lines of responsibility and authority between systems engineering and program management.
SP 2.1-5-2a	Assign responsibility for program planning.
SP 2.1-5-2b	Include in the technical management plan provisions to maintain the plan and for recording deviations from the plan.
SP 2.1-5-2c	Document the program roles, responsibilities, and objectives for each organization or functional discipline.
SP 2.1-5-2d	Develop an event driven plan for technical aspects of the program.
SP 2.1-5-2e	Review technical plans with stakeholders and obtain their commitment.
SP 2.1-5-3a	Ensure the systems engineering activities and work products that are needed to establish and maintain control of the program are well defined.
SP 2.1-5-3b	Conduct formal reviews of the technical management plan to assess its consistency with the top-level program management plan and with lower-level plans.

FA 2.2 Monitor and Control

*A Focus Area for the Systems Engineering **Management** Category*

Description:

Monitor and Control provides visibility into program progress and risks, and detection of variances needing corrective action. The monitor aspect requires tailoring the level of control to the complexity and risk of the program, tracking data for that level of control, and initiating corrective action when measures do not meet expected results. The control aspect requires setting thresholds of control limits and activating corrective actions based on risk analysis. The review process compares the results against the program's documented estimates, commitments, and plans. Adequate visibility enables timely corrective action to be taken before performance deviates significantly from plans.

Monitoring includes the measurement of program functions and tasks identified in systems planning. Technical characteristics, program progress (cost, schedule), and systems engineering process activities are tracked.

Individual parameters to be measured will have been determined by policy, plans or preceding effort. These measures provide the information necessary to synthesize metrics, which generically include technical parameters for the product, program progress, and systems engineering process activities.

Control consists of the initiation of corrective action as required. Control actions mitigate issues or accept risk. Each of these control actions should balance the organization's needs and goals with those of the program, and should be communicated to all stakeholders.

Technical performance metrics are used to track key technical parameters throughout a development program. Planning and control metrics provide a periodic assessment of the health and status of the program throughout the life cycle. Systems engineering process metrics provide an indication of the quality and productivity of the systems engineering process as applied to a specific program.

Notes:

This Focus Area addresses issues pertaining to the systems engineering technical effort at a program level but will mention applicability to the organizational level where appropriate. Rationale for appropriate application of this FA should be established prior to program initiation or during program planning. The Monitor and Control effort includes elements of the program's environment, including:

- Computing resources,
- Systems engineering tools,
- Production productivity tools,
- Communications tools,
- Analysis methods,
- The organization's policies and procedures,
- Machine shops,
- Chemical,
- Environment stress facilities, and
- Work space.

References:

Not Applicable.

Themes of Monitor and Control:

2.2-1 Degree of Formality
2.2-2 Monitoring
2.2-3 Thresholds Exceeded

Theme Descriptions, Typical Work Products, and Practices:

2.2-1 Degree of Formality

Description:

The resources applied to the Monitor and Control effort depend on the degree of formality established. The formality of the control has a relationship to cost, schedule, and performance risk. The organization's planning and control needs are considered when planning the effort, as are supporting policies, procedures, and processes.

Typical Work Products:

- Input to program management and control structure
- Budget for monitor and control efforts

Specific Practices:

SP 2.2-1-1 Determine the degree of oversight for programs needing monitoring and controlling to promote the organization's goals.

SP 2.2-1-2 Establish criteria against which each program is evaluated to determine if it should be under the Monitor and Control FA activities.

2.2-2 Monitoring

Description:

Reviews of technical parameters, plans, schedule issues and documented program history provide information for either a correctly operating process or when an abnormal condition occurs.

Typical Work Products:

- Measures to evaluate the effectiveness of the processes
- Reports comparing the program status to plans
- Statistical techniques analysis results in the form of graphs, run charts, pareto charts, forcefield analysis, etc.
- Report on duration of use of tools (automated and manual)
- Threshold memo
- Change Request (design, process, specification, requirement, etc.)
- Trend charts showing cycle time for development
- Technical reviews (e.g., requirements reviews, design reviews, test reviews)

Specific Practices:

SP 2.2-2-1	Track the resources expended, the program schedule, and the technical performance measurements against the plan.
SP 2.2-2-3	Evaluate and document the program’s efforts for the lessons learned.

2.2-3 Thresholds Exceeded

Description:

Control of a program requires knowing when and how to apply corrective action. This theme covers data collection, analysis, and initiation of corrective action. When required, corrective action reduces special and variant causes and prevents deviation(s) from planned events. When the program’s process has deviated from planned performance or organizational guidelines, the organization’s policies and plans dictate the course of action.

Comments:

Measures may be derived from technical performance, schedules, cost, or any planned event.

Typical Work Products:

- Revised program completion schedule
- Plans for specific corrective actions to mitigate the program’s deviations
- Recommendations for program termination
- Recommendations for personnel re-allocation
- Recommendation on rescheduling program completion
- Lessons learned documentation
- Plans for recovery of program direction
- Action items list

Specific Practices:

SP 2.2-3-1a	Periodically collect and analyze the measures of program and technical performance.
SP 2.2-3-1b	Implement corrective action when measures deviate from expected results.
SP 2.2-3-4	Analyze and use prediction based on the program’s measures to determine if the program’s completion is at risk and thus warrants corrective action.

FA 2.3 Integrate Disciplines

*A Focus Area for the Systems Engineering **Management** Category*

Description:

Integrate Disciplines selects and blends the interdisciplinary people necessary for system development such that they jointly and effectively work together. Organizational and information structures and communication processes are put in place that enable each disciplines unique expertise and concerns to be identified, considered, and focused on system engineering. A key element in the integration of disciplines is the anticipation of conflicts and the establishment of a process for resolution of issues.

An organization's concurrent engineering practices, interdisciplinary teaming, and integrated product development process may meet the requirements of this Focus Area if they include the practices described below. For example, a typical approach might be to integrate disciplines into a team where team members share responsibility for product development and delivery.

Notes:

In the context of identifying system stakeholders, as described in EIA Standard 632, *Processes for Engineering a System* [1], the disciplines addressed in this Focus Area are mainly those that fall within the category of system developer. These disciplines typically include application-domain specialists, marketing, manufacturing, and all engineering disciplines necessary for system and component design, development, reliability, maintainability, quality, support and logistics, human factors, safety, and security. This integrated environment should continue throughout a system's life cycle. Stakeholders other than the developer may be integrated into the development process, i.e., many successful system development efforts involve suppliers, customers, and users throughout the development process, and involve developers in the manufacturing, installation, and operation of the system.

References:

1. EIA Standard 632, *Processes for Engineering a System.*

Themes of Integrate Disciplines:

2.3-1 Involvement of Affected Individuals
2.3-2 Understanding, Communication, and Coordination
2.3-3 Interdisciplinary Issue Resolution

Theme Descriptions, Typical Work Products, and Practices:

2.3-1 Involvement of Affected Individuals

Description:

Efficient and effective system development results from a blending of the efforts of people from many disciplines. The needed people should be identified during planning and ideally involved in the planning effort. They will be involved in the development effort as dictated by the program's milestones and development schedule. Concurrent engineering principles and best practices suggest that product or component designers, builders, and testers should be involved in the system definition process.

Comments:

The use of integrated product teams to meet the needs for interdisciplinary cooperation is a common practice for creating a concurrent engineering environment. Teams should be formed to align with the product structure, with each team held responsible for planning, developing, and satisfying the requirements associated with its work package(s). Concurrent engineering ensures that all product stakeholders - manufacturers, users, maintainers, etc. - have a say in the system requirements, design, and modifications.

Typical Work Products:

- Information Management Plan
- Computer-Aided Engineering Plan
- Program Staffing Plan, including a roster of essential disciplines and their representatives
- Program Management Plan, including agendas and schedules for collaborative activities
- Organizational Breakdown Structure with responsibilities and authorities assigned

Specific Practices:

SP 2.3-1-1	Involve all essential disciplines, including both traditional and specialty engineering, in the system development process in a timely manner.
SP 2.3-1-2	Adjust the mix of disciplines involved in each phase of system development as appropriate to the work being done.
SP 2.3-1-3a	Involve personnel from affected groups in planning and other systems engineering activities (i.e., developing, reviewing, allocating, and approving requirements) that affect them.
SP 2.3-1-3b	Allow systems engineering personnel to review and agree to designs, plans, and work products produced by other engineering disciplines or that affect multiple disciplines.

2.3-2 Understanding, Communication, and Coordination

Description:

Leadership is required to ensure that various engineering and other disciplines come together and collaborate on a system development effort. Engineering disciplines typically each have their own models, tools, languages, and method-s that are focused on development of a part of the system and that need to be integrated into a system development process. Proactive management ensures that there is good understanding of requirements, technologies, risks, trade offs, interfaces, and issues that permeate the development effort. Systems Engineering’s models, tools, and information should be open, accessible, and understandable to all disciplines. Efforts should be made to ensure that each discipline understands the other disciplines contributions and importance to the development effort.

Understanding of the relationships between disciplines is important. This involves identifying when and how information is exchanged, who sits where, who is notified when information is developed and changed and when decisions are weighed and made.

Comments:

Knowledge sharing may center around an automation strategy, in which case individuals would share knowledge through the automation tool suite, or knowledge sharing may center around a teaming strategy, in which case individuals would share knowledge through the particular teaming structures used.

Typical Work Products:

- Integrated information infrastructure
- Communications via inter/intra-net
- Integrated product teams

Specific Practices:

SP 2.3-2-1	Proactively emphasize the importance of intergroup coordination.
SP 2.3-2-2a	Capture and communicate intergroup coordination activities and the results of those activities.
SP 2.3-2-2b	Establish tools, methods, facilities (e.g., team rooms), and an information infrastructure that eases and supports interdisciplinary coordination.
SP 2.3-2-3a	Provide means for individuals and groups to acquire skills that facilitate interdisciplinary cooperation, such as communication skills, group problem solving, and active listening.
SP 2.3-2-3b	Plan for and provide regular exchanges of technical information and issue identification and resolution among all stakeholders, including customers.
SP 2.3-2-3c	Establish a mechanism to ensure compliance with commitments made among groups.
SP 2.3-2-4	Espouse and model appropriate communication skills and interdepartmental cooperation on the part of upper management.

2.3-3 Interdisciplinary Issue Resolution

Description:

System development is characterized by decisions that seek a balanced interdisciplinary resolution to system-level problems. Inherent in that principle is the fact that, in many cases, decisions must be made that do not allow a solution that seems locally or immediately optimal (such as choice of a particular part or technology, or inclusion of features or efforts that seem to cost more or take more time in the short run). Such decisions can generate conflict, as can unforeseen difficulties, changing customer needs and market environment, etc. Anticipation of such problems is inherent in a good systems engineering process.

Comments:

Methods for conflict resolution should be identified during the program planning activities which are defined in FA 2.1 - Plan and Organize.

Typical Work Products:

- Meeting minutes
- Integrated decision database
- Trouble reporting system

Specific Practices:

SP 2.3-3-1	Establish and use a process or method for identifying and resolving interdisciplinary issues.
SP 2.3-3-2	Communicate interdisciplinary issues and activities to affected groups, including program/project management and customer, supplier, and associate stakeholders.
SP 2.3-3-3	Establish a process for escalating and arbitrating technical differences, including a mechanism for authoritative resolution of conflicts.

FA 2.4 Coordinate with Suppliers

*A Focus Area for the Systems Engineering **Management** Category*

Description:

Coordinate with Suppliers addresses the needs of the organization and programs to effectively select and manage those portions of product work that are conducted by outside suppliers. The general term ‘supplier’ is used to identify an internal or external outside organization that develops, manufactures, tests, or supports a component of the system. Suppliers may take many forms including in-house vendors, fabrication capabilities and laboratories, and partners depending on business needs.

Coordination with suppliers involves (1) identifying work that will be outsourced, (2) determining suppliers capable of accomplishing that work in an effective manner, (3) defining an agreement or contract with the supplier, and (4) managing to that agreement.

Affected organizations must have a shared vision of the working relationship in addition to coordination of schedules, work processes, and the deliveries of work products. Relationships can range from integrated developer-supplier product teams, to prime-contractor/subcontractor, to vendors, etc. A successful relationship between an organization and a supplier depends on the capabilities of both organizations, as well as a mutual understanding of the relationship and expectations.

Notes:

Decisions made as a part of this Focus Area should be made in accordance with FA 1.4 - Assess and Select. The organization acts as the customer when the supplier executes the activities associated with FA 1.1 - Define Stakeholder and System Level Requirements. Activities associated with managing the supplier should be in accordance with FA 2.2 - Monitor and Control.

References:

Not Applicable.

Themes of Coordinate with Suppliers:

2.4-1 Outsourced Work Product Identification
2.4-2 Supplier Selection
2.4-3 Work Product and Activity Definition
2.4-4 Supplier Management

Theme Descriptions, Typical Work Products, and Practices:

2.4-1 Outsourced Work Product Identification

Description:

Rarely does an organization make every component of a system. The organization decides the components that will be procured from an outside source and which components will be developed in-house, based on determining core competencies and trade studies. The organization conducts make-vs-buy analyses and determines those items to be procured. System needs that will be satisfied outside the organization fall generally into two categories: (1) components that the organization has little expertise or interest, and (2) components that can be procured more economically than they could be developed in-house.

Typical Work Products:

- Make-vs-buy decisions
- Trade studies
- Sub-set of system components for external purchase
- Sub-set of system components for internal developments/fabrication

Specific Practices:

SP 2.4-1-1	The organization identifies system components or services that will be provided by internal and external suppliers.
SP 2.4-1-2	The organization performs trade studies to determine make-vs-buy decisions based on business needs.

2.4-2 Supplier Selection

Description:

The selection of an appropriate supplier(s) has a major impact on the success of the overall program. The capabilities of the supplier should be complementary and compatible with those of the organization. Issues that may be of concern include competent development processes, manufacturing processes, responsibilities for verification, on-time delivery performance, life-cycle support processes, and ability to communicate effectively.

Suppliers are selected in a logical manner to meet program and organization objectives. The characteristics of a supplier that would best complement the organization's abilities are determined, and qualified candidates are identified. The activities of FA 1.4 - Assess and Select are applied to select the appropriate supplier.

Comments:

An important consideration in the selection of the supplier is the expected working relationship. This could range from a highly integrated product team to a classical "meet the requirements" relationship. The selection criteria differ, depending on the anticipated relationship.

Typical Work Products:

- List of candidate suppliers
- Advantages and disadvantages of supplier being considered
- List of organizational weaknesses which may be mitigated by an external supplier
- Characteristics of the desired working relationships with the supplier
- Customer requirements to be "flowed down" to supplier
- Preferred supplier list

- Capture of rationale for selection of suppliers

Specific Practices:

SP 2.4-2-1	Capable suppliers are chosen according to FA 1.4 - Assess and Select.
SP 2.4-2-2a	Criteria are established to evaluate potential suppliers that meet program and organization objectives.
SP 2.4-2-2b	Suppliers are selected based upon input from the systems engineering team leader.
SP 2.4-2-3	Suppliers are selected based on an evaluation of the supplier's ability to perform the work.

2.4-3 Work Product and Activity Definition

Description:

The program and organization clearly identify and prioritize their needs and expectations, as well as any limitations on the part of the suppliers. The organization works closely with suppliers to promote a mutual understanding of product requirements, responsibilities, and processes to achieve program objectives. The relationship between the acquirer and supplier is documented in a contract or other agreement.

Typical Work Products:

- Needs and expectations statement
- Subcontract(s) or other agreements
- Statement(s) of work
- Subcontract specifications
- Technical input to Subcontract Management Plan
- Technical performance parameters
- Measures of effectiveness
- Verification specifications
- Acceptance criteria

Specific Practices:

SP 2.4-3-1	The organization provides the supplier with the needs, expectations, and measures of effectiveness for the system components and services to be delivered.
SP 2.4-3-2a	When suppliers are used on the program, requirements for the work are formally documented.
SP 2.4-3-2b	Requirements changes are re-negotiated with the supplier and the changes documented.
SP 2.4-3-3	There is a clearly documented agreement that contains a statement of work, specification, terms and conditions, a list of deliverables, a schedule, budget, and a defined acceptance process.
SP 2.4-3-4	The selected supplier is involved early in the program to assist in the requirements development and definition.

2.4-4 Supplier Management

Description:

Supplier management involves monitoring and controlling the activities of the supplier. The level and methods of supplier control will be based on the relationship established and the significance of the component to the overall system. The program and organization review and concur with the supplier's plans and procedures. The program should also monitor the supplier's conformance to these plans and procedures through appropriate reviews and audits. Reviews include configuration control, quality, technical performance, costs and schedule. Appropriate acceptance testing ensures that the delivered component satisfies the need.

Typical Work Products:

- Supplier subcontract progress reports
- Formal reviews
- Audit reports
- Results of acceptance testing

Specific Practices:

SP 2.4-4-1	The supplier's progress (schedule, cost, technical performance) is managed.
SP 2.4-4-2a	Those involved in managing the supplier receive orientation in the technical aspects of the documented agreement.
SP 2.4-4-2b	The supplier's quality and configuration control activities are monitored.
SP 2.4-4-2c	Acceptance testing is conducted as part of the delivery of the supplier's products.
SP 2.4-4-3a	The documented agreement between the acquirer and the supplier is used as the basis for managing the supplier.
SP 2.4-4-3b	Periodic informal reviews, technical reviews, and interchanges are held with the supplier.
SP 2.4-4-3c	Formal reviews are conducted at selected milestones to address the supplier's systems engineering accomplishments and results.
SP 2.4-4-3d	Discrepancies discovered during acceptance testing are used to improve the supplier's processes and products.
SP 2.4-4-3e	There is a mechanism for assuring that all suppliers follow their defined engineering process.
SP 2.4-4-4a	Systems engineering personnel participate in and approve the plans, process, and product standards used by suppliers.
SP 2.4-4-4b	There is a mechanism for establishing and nurturing long term relationships with preferred suppliers.

FA 2.5 Manage Risk

*A Focus Area for the Systems Engineering **Management** Category*

Description:

Manage Risk involves identifying, assessing, monitoring, and mitigating risks for the performance (i.e., technical, programmatic, supportability), cost and schedule activities of a program throughout the life-cycle [1]. While technical issues are a primary concern for systems engineering during a program, the three elements (i.e., performance, cost, schedule) must be balanced for a successful risk management process and program.

Risk management involves five related activities: planning for risk, risk identification (sometimes called risk assessment), risk analysis, risk mitigation (sometimes called risk handling), and communication. Risk identification is the continuous process of identifying all areas of potential risk on a program. The thoroughness with which risk identification is done is a key factor in the effectiveness of risk management; if risks are not identified then analysis and appropriate corrective action cannot be taken. Risk analysis is the process of quantifying each specific risk; determining the probability of its occurrence and the impact on the program associated with its occurrence; and developing and analyzing alternative options, backed up by specific recommendations for action. Risk mitigation is the process of avoiding, reducing and controlling, or deliberately accepting risk on the program. In order to be effective, risk management activities need to be communicated to all affected groups and individuals.

Risk management is both a program and technical management process for a program conducted in an uncertain environment. Risk management provides a means of addressing issues which have the potential for causing a specified performance, schedule, or cost requirement on the program to not be satisfied. Risk management should not be viewed as a separate program office function, but rather as an integral part of sound systems engineering management of the technical effort. In essence, risk management is “a method of managing that concentrates on identifying and controlling the areas or events that have a potential of causing unwanted change and it is no more and no less than informed management” [2].

Risk management is considered a critical systems engineering activity; yet effective risk management is one of the most difficult and elusive tasks [3]. The effectiveness of the program risk management effort will be based upon the breadth of experience, depth of system expertise, interface and integration experience, and creativity of the assigned risk manager, program team, and the participants from the line organizations. This will be further influenced by the scope of the charter provided by the program manager, whose support cannot be underestimated for this process to be beneficial and timely.

Identification, analysis, and mitigation of risks early in the product life cycle is important because typically 70 percent of life cycle costs are committed, or “locked in”, by decisions made during the expenditure of the first 5 percent of the life cycle cost [4]. In each product life cycle phase, early detection of risk is important because it is easier and cheaper to make changes and correct errors than in the middle or the end of that phase [5].

Risk management applies to all program sizes with additional considerations taken into account when implementing the process. For instance, the formality depends on the program size, program phase, funds available, program complexity, and maturity of the technology being considered. Small programs may not need a very formal approach, but regardless of the program size, the approach requires discipline and should include all program suppliers.

The risk management approach selected for the Manage Risk Focus Area is oriented about the approach taken to address risks developed by the Defense Systems Management College (DSMC), which has become widely used within industry. A somewhat useful, but earlier source, on risk management from DSMC is given in [6].

Notes:

The risk management strategy developed by FA 2.5 - Manage Risk must be incorporated into the technical management plan, if only by reference, developed under FA 2.1 - Plan and Organize. The risk management approach is updated by the activities of FA 2.2 - Monitor and Control. Risk management activities assist in the selection of suppliers and the control of their activities under FA 2.4 - Coordinate With Suppliers.

References:

1. *Risk Management: Concepts and Guidance*; page 3-3; Defense Systems Management College (Fort Belvoir); March 1989.
2. Caver, T. V.; "Risk Management as a Means of Direction and Control"; *FAPt Sheet Program Managers Notebook*; Defense Systems Management College (Fort Belvoir), No. 6.1; April 1985.
3. *Systems Engineering Management Guide*; page 12-9; Defense Systems Management College (Fort Belvoir); May 1996 (Draft.
4. *Systems Engineering Management Guide*; pages 17-1 through 17-3; Defense Systems Management College (Fort Belvoir); January 1990.
5. Hudak, G. J., et al., *Design to Reduce Technical Risk*, page 9, McGraw-Hill, 1993.
6. *Risk Assessment Techniques*; Defense Systems Management College (Fort Belvoir); July 1983.

Themes of Manage Risk:

2.5-1 Risk Management Plan
2.5-2 Identification of Performance, Cost, and Schedule Risks
2.5-3 Risk Quantification
2.5-4 Risk Analysis
2.5-5 Development of a Risk Mitigation Strategy
2.5-6 Implementation of the Risk Mitigation Strategy
2.5-7 Monitoring of Risk Mitigation Action
2.5-8 Communication and Coordination of Risk Status and Risk Mitigation Efforts Across Affected Groups

Theme Descriptions, Typical Work Products, and Practices:

2.5-1 Risk Management Plan

Description:

Most programs are guided by an overall, encompassing technical management plan that includes, sometimes by reference, a series of sub-plans. A risk management plan is an essential part of this suite of sub-plans. The risk management plan should be developed at the beginning of the program and updated as needed throughout the life of the effort.

The purpose of a risk management plan is to force an organized approach to the subject of eliminating, minimizing, or containing the effects of undesirable occurrences. The risk management plan should contain the essentials of any plan (see Plan under Glossary), but in addition, address those essential items of the risk effort. Some of these essential items that characterize the risk management effort are the approach to: identifying, quantifying, and analyzing risks; developing, implementing, and monitoring the risk mitigation activities; and communicating and coordinating the risk management activities across affected groups.

Comments:

Risk level (derived using a risk model) is a measure combining the uncertainty of reaching a goal with the consequences of failing to reach the goal.

Typical Work Products:

- Risk management plan
- Risk model

Specific Practices:

SP 2.5-1-1	Plan risk management activities.
SP 2.5-1-2	Provide an approved risk management plan containing risk levels and expected management response for each level.
SP 2.5-1-3	Implement risk management for key processes within the program: design, test, manufacturing, etc.

2.5-2 Identification of Performance, Cost, and Schedule Risks

Description:

Risks must be identified, and described in an understandable way, before they can be managed. Risk identification should be an organized, thorough approach to seek out probable, or realistic, risks in the program associated with performance, cost, and schedule. To be effective, it should not address highly improbable events in the program.

Comments:

There are many methods for identifying risks. One method to ensure that all program risks have been identified is to examine each element of the program work breakdown structure to uncover risks. Other methods of identifying program risks are: interview subject matter experts, review similar system risk management efforts, examine lessons learned documents or databases.

Typical Work Products:

- List of program risks

Specific Practices:

SP 2.5-2-1	Identify performance risks.
SP 2.5-2-2	Identify cost and schedule risks.
SP 2.5-2-3	Review all elements of the work breakdown structure as part of the risk identification process in order to help ensure that all program aspects have been considered.

2.5-3 Risk Quantification

Description:

After a list of risks have been generated and before risk analysis begins in earnest, the identified risks are quantified. The quantification of each risk has two aspects: the potential that something will go wrong combined with the consequences should it will go wrong.

Quantification of risk for each of the above two aspects can be done in a three-tier approach in terms of “high”, “medium”, and “low”. This approach is often appropriate in a system involving a small number of elements that are simply related. The three-tier approach tends to provide a top-level intuitive approach to addressing risk. It is also an approach often favored in management reviews where a red-yellow-green notation is used and corresponds to high-medium-low, respectively.

Alternatively, for systems involving large numbers of elements that are interrelated in a complex manner, it is often more appropriate to establish a mathematical model to describe risk for the program. This mathematical model expresses the identified risks, in terms of their above two aspects, and the interrelationships using the mathematics of probability.

Typical Work Products:

- List of risks for the program, with a risk rating (probability of occurrence and consequence of impact) for each risk

Specific Practices:

SP 2.5-3-1	Assess risks qualitatively.
SP 2.5-3-2	Assess each risk and determine the probability of occurrence and quantified consequence of impact for the program.

2.5-4 Risk Analysis

Description:

Risk analysis involves conducting an analysis of all aspects of the program to determine the logical relationships of cause and effect for each identified risk, the magnitude of the risk, its effect on the program, and determination of alternative options for handling each risk. A "watch list" is often generated that identifies each risk, consequences of the risk (often for high and medium risks only), indicators that signal (trigger) the start of a problem(s) indicative of the risk, related area of impact on the program, and risk handling actions that can be taken to avoid/minimize the risk.

Typical Work Products:

- Risk watch list
- Cumulative risk probability distribution

Specific Practices:

SP 2.5-4-3a	Review the analysis of risks for adequacy and completeness.
SP 2.5-4-3b	For each risk, establish cause and effect relationships.
SP 2.5-4-3c	Analyze each risk for potential coupling to all other identified risks.
SP 2.5-4-3d	Develop alternative courses of action, work-arounds, and fall-back positions with a recommended course of action for each risk.
SP 2.5-4-4	Use collected metrics regarding identified risks and examine them in light of previous risk analyses, and when established thresholds are exceeded, initiate corrective action.

2.5-5 Development of a Risk Mitigation Strategy

Description:

A risk mitigation strategy, also known as risk handling, is developed to handle each of the risks identified. Quite often, risk handling is only performed for those risks judged to be "high" and "medium". The risk mitigation strategy for a given risk includes techniques and methods to avoid, reduce and control the probability of occurrence of the risk or the extent of damage incurred should the risk (anticipated event or situation) occur or both. Often, especially for "high" risks, more than one approach to mitigating a risk will be generated.

Not all risks can be completely mitigated. In some cases, mitigation of a particular risk simply transfers the risk to another area, hence, the need for performing an effective risk analysis.

In some cases, the strategy may be to simply accept the risk. Risk acceptance is usually done when the risk is judged to be "low" or when there does not appear to be a viable way to reduce or control the risk.

It should be noted that the absence of a risk mitigation strategy means that risk is not being managed, only observed.

Typical Work Products:

- Risk mitigation strategy

Specific Practices:

SP 2.5-5-2	Categorize risks into those that can be avoided, controlled, or accepted.
SP 2.5-5-3a	Document risk reduction profiles and review them for appropriateness.
SP 2.5-5-3b	Review risk mitigation (handling) including risk reduction profile for adequacy and completeness.

2.5-6 Implementation of the Risk Mitigation Strategy

Description:

The mitigation strategy for appropriate risks needs to be implemented to ensure that the risk (potential occurrence and impact) does not occur. Implementation of a risk mitigation strategy may mean reducing either the probability that a risk will occur or extent of damage caused if the risk occurs or both.

Typical Work Products:

- Risk management plan
- Risk mitigation strategy
- Watch list
- Risk Reviews

Specific Practices:

SP 2.5-6-2	Implement the risk mitigation strategy for the program.
SP 2.5-6-3	Document risk analysis results and mitigation plans.

2.5-7 Monitoring of Risk Mitigation Action

Description:

To effectively control risk over the duration of the program means regularly monitoring the results of the risk mitigation actions. This means continually sensing the condition of the program; determining the degree of success of existing risk mitigation actions; developing new options and fall back positions for risk mitigation actions that have not achieved the required effect; and identifying new risks, quantifying and analyzing these risks, and developing and implementing appropriate mitigation strategy(s).

Typical Work Products:

- Updates to risk management plan
- Updates to list of program risks

- Updates to risk ratings
- Updates to risk watch list
- Updates to cumulative risk probability distribution

Specific Practices:

SP 2.5-7-3a	Monitor and re-evaluate risks at appropriate milestones.
SP 2.5-7-3b	Provide the results of risk monitoring activities to affected personnel and disciplines.
SP 2.5-7-3c	Provide a mechanism for monitoring corrective actions taken and tracking open risk items to closure.
SP 2.5-7-4	During risk monitoring, identify and analyze new risks and take corrective action.

2.5-8 Communication and Coordination of Risk Status and Risk Mitigation Efforts Across Affected Groups

Description:

Risks cannot be effectively managed unless all affected groups are aware of, and participate in, the risk management activities.

Typical Work Products:

- Briefings
- Correspondence
- Status Reports

Specific Practices:

SP 2.5-8-1	Establish a communication path between the risk management team and the program management team.
SP 2.5-8-2a	Involve a multi-functional group for risk management that spans both technical and business specialties.
SP 2.5-8-2b	Integrate risk management both vertically and horizontally across the program.
SP 2.5-8-3	Include risk management as a part of program formal reviews.

FA 2.6 Manage Data

A Focus Area for the Systems Engineering ***Management*** *Category*

Description:

Data management is administrative control of program data, both deliverable and non-deliverable. Administrative control involves such activities as identification, inspection, maintenance and distribution, change control, statusing, and retrieval.

Data are the various forms of documentation required to support a program in all of its areas (e.g., administration, engineering, configuration, financial, logistics, quality, safety, manufacturing, and procurement). The data may take any form (e.g., reports, manuals, notebooks, charts, drawings, specifications, files, or correspondence). The data may exist in any medium (e.g., printed or drawn on various materials, photographs, electronic, or multi-media). Data may be deliverable (e.g., items identified by a program's contract data requirements), or data may be non-deliverable (e.g., informal data, trade studies and analyses, internal meeting minutes, internal design review documentation, lessons learned and action items). Distribution may take many forms including electronic transmission.

Data Management begins at the initial program stage, prior to any proposal activities, and continues throughout the life of the program.

Notes:

1. This Focus Area interfaces with FA 2.1 - Plan and Organize. The key to data management is the planning of the data products that are required, the actions necessary to create and make available that data, and the responsibilities for those actions. The identification theme is interrelated with the program planning as it includes the definition of data and data related tasks. Status, as used in this Focus Area, depends on the proper accomplishment of that planning.

2. FA 2.6 - Manage Data and FA 2.7 - Manage Configurations are interrelated. Both include control of content, versions, changes, and distribution of data. Data Management addresses data only and addresses data development schedule requirements. Configuration Management is more focused on the control of the technical aspects of the product and also includes the delivered system.

3. The means of management, control and distribution, particularly in an electronic environment, must be coordinated with FA 3.4 - Manage Systems Engineering Support Environment.

References:

Not Applicable.

Themes of Manage Data:

2.6-1 Data Requirements Identification
2.6-2 Inspection
2.6-3 Maintenance and Distribution
2.6-4 Change Control
2.6-5 Status

Theme Descriptions, Typical Work Products, and Practices:

2.6-1 Data Requirements Identification

Description:

The data requirements for the program should be established for both the data items to be created and their content and form. This task includes the analysis and validation of program deliverable and non-deliverable, contract and non-contract data requirements, subcontractor, associate contractor, and customer supplied data.

Typical Work Products:

- Data Management Plan (DMP)
- Contract Data Requirements List (CDRL)
- Data Delivery Schedules
- Data Accession List
- Subcontract Data Requirements List

Specific Practices:

SP 2.6-1-1 Establish program data and data management requirements.

SP 2.6-1-2 Review data management activities periodically to confirm that the program data requirements are still valid and on schedule.

SP 2.6-1-3 Establish program data requirements based upon a common or standard set of data requirements.

2.6-2 Inspection

Description:

Program data should be inspected for compliance to data requirements prior to delivery or archiving. The inspection task includes interpretation of data requirements.

Specific Practices:

SP 2.6-2-1 Inspect program data for compliance to data requirements prior to delivery or archiving.

SP 2.6-2-2 Ensure that the process for review, approval and release of data is well understood through the program.

2.6-3 Maintenance and Distribution

Description:

All data products should be received, logged, archived, recovered, transmitted, and distributed as required.

Comments:

The archival system provides capture techniques for program data that are appropriate to the degree of formality of the data. It includes processes for retrieving, storing, and disposing data.

Specific Practices:

SP 2.6-3-1 Archive program data.

SP 2.6-3-2 Establish a capability to retrieve desired program data quickly.

SP 2.6-3-3a Provide a common data management archival and retrieval capability throughout the organization.

SP 2.6-3-3b Archive data efficiently based upon common characteristics (e.g., key words, topics, contract number, etc.).

2.6-4 Change Control

Description:

Data item changes are controlled and evaluated consistent with the technical, program, and configuration management requirements.

Specific Practices:

SP 2.6-4-1	Changes to data requirements and data are controlled and communicated.

2.6-5 Status

Description:

The status activity will identify all data item requirements, their individual delivery schedules, and the performance history against those schedules. Information may identify originators of each data product, customer response dates, disposition, file page counts, security classification, etc. The status of the program data is is communicated to those that need it.

Specific Practices:

SP 2.6-5-1	Record and maintain the status of program data.
SP 2.6-5-2a	Communicate status reports documenting data management activities to appropriate groups and individuals.
SP 2.6-5-2b	Alert individuals having responsibility for the generation of program data of upcoming milestones and delivery dates.

FA 2.7 Manage Configurations

*A Focus Area for the Systems Engineering **Management** Category*

Description:

Configuration Management involves configuration identification, change control, status accounting, and auditing of the product and product elements which define the product. These include requirements, interfaces, and design representations of the products being provided to meet the stated program objectives.

Manage Configuration assures that the customer has the correct product and that the developer maintains data and status of identified configuration items, and can analyze and control changes to the system and its configuration items. Managing the system configuration involves providing accurate and current configuration data and status to developers and customers.

This Focus Area is applicable to all work products that are placed under configuration management. Examples of work products that may be placed under configuration management include hardware and software configuration items, design rationale, requirements, product data files, and trade studies.

Notes:

FA 2.7 - Manage Configurations and FA 2.6 - Manage Data are interrelated. Both include control of content, versions, changes, and distribution of data. Configuration Management is more focused on the control of the technical aspects of the product and also includes the delivered system. Data Management addresses data only and addresses data development schedule requirements.

References:

1. EIA STD 649, *National Consensus Standard for Configuration Management.*

Themes of Manage Configurations:

2.7-1 Identification
2.7-2 Change Control
2.7-3 Status Accounting
2.7-4 Audit

Theme Descriptions, Typical Work Products, and Practices:

2.7-1 Identification

Description:

Configuration identification is the selection, creation, and baselining of items that comprise the system and the documents that describe the system. Items under configuration management will include specifications and interface documents that define the requirements for the product. Other documents, such as test results, may also be included depending on their criticality to defining the product.

Baselining is the act of placing an item under change control such that all changes are made through a defined process. The baseline configuration defines the characteristics of an item during its life cycle.

Comments:

Multiple baselines may be used to define an evolving product during its development cycle. One common set includes the system level requirements, system element level design requirements, and the product definition at the end of development/beginning of production. These are referred to as the functional, allocated, and product baselines.

Baselines may also be established at different times internally and with a customer. An internal baseline of drawings might be established before fabrication with the customer baseline occurring at delivery.

Optimization includes defining the correct items to manage and establishing the correct timing for baselines. Baselining too soon (before maturity starts) increases the administrative costs of changes, and baselining too late increases the risk of undesired changes.

Typical Work Products:

- Baselined work product configuration
- Identified configuration items
- Decision database
- Baselined configuration
- Traceability matrix
- Specification trees
- Drawing trees
- Release documents
- Specifications
- Interface control documents
- Configuration Management Plan

Specific Practices:

SP 2.7-1-1 Identify, baseline, and control work products that define the product.

Note: Although the responsibility for technical content of the products remains with the systems engineering technical task, the configuration management identification function shares responsibility for proper structure of specification trees, level of documentation, and general content and format of documentation.

SP 2.7-1-2a Identify, baseline, and control work products from all Focus Areas that are critical enough to require configuration management.

SP 2.7-1-2b Maintain a repository of work product baselines.

SP 2.7-1-2c Maintain the capability to store, manage, retrieve, and distinguish multiple versions of product elements and work products.

SP 2.7-1-3 Formally control release of products created from the baseline library.

2.7-2 Change Control

Description:

Configuration change control is the control of changes to baselined items through recording, review, and approval processes.

Comments:

Optimization of this theme includes the selection of the level of formal control to be implemented. A balance is needed between the cost of formal control and the risk of uncontrolled changes.

Typical Work Products:

- Change requests (CR)
- Specification change notices
- Document change notices
- Decision records
- Modified work-product baselines
- Deviations and Variances
- Waivers

Specific Practices:

SP 2.7-2-1 Changes to established baselines are recorded, reviewed, approved, controlled, and verified as incorporated.

SP 2.7-2-2 Changes are evaluated through a process that ensures they are consistent with all the technical and program requirements.

SP 2.7-2-3 Changes are evaluated for their impact beyond the immediate program or contract requirements.

Note: Changes to an item used in multiple products may resolve an immediate issue and cause a problem in other applications.

2.7-3 Status Accounting

Description:

Configuration status accounting provides the recording and reporting of change information to the baselined configuration items. It provides the traceability of configuration identification and facilitates the effective implementation of approved changes.

Typical Work Products:

- Configuration status accounting reports
- As built list or records
- Release notes

Specific Practices:

SP 2.7-3-1 Status of configuration data, changes, and access information is recorded, tracked, and communicated to affected groups.

Note: Examples of activities for communicating configuration status include providing access permissions to authorized users and making baseline copies readily available to authorized users.

2.7-4 Audit

Description:

Configuration auditing involves the checking of an item for compliance with its configuration baseline and the accuracy of the baseline documentation. Configuration audits validate that the developed item fulfills its technical requirements, e.g., in a Functional Configuration Audit (FCA), and that the product configuration is properly identified, e.g., in a Physical Configuration Audit (PCA). This is accomplished by comparing the configuration item with its technical documentation and the status reports.

Typical Work Products:

- Audit results
- Action items

Specific Practices:

SP 2.7-4-1 Periodically audit configuration management activities and processes to confirm that the resulting baselines and documentation are accurate and record audit results.

Note: Audits should confirm both the accuracy and currency (incorporation of changes, etc.) of each level of baseline (requirements/design/product) and the consistency between levels (requirements match the design and the design matches the product baseline).

FA 2.8 Ensure Quality

*A Focus Area for the Systems Engineering **Management** Category*

Description:

Ensure Quality addresses the quality of the system and the quality of the process(es) being used to create the system. A high-quality system can only be produced, on a continuous basis, if a process exists to continuously measure and improve the quality of the processes used to produce the system products. This process emphasizes establishment of quality goals and subsquent measurements, analysis, and implementation of corrective action to attain the goals. It encompasses themes of:

- **Leadership and Involvement** — Strong and effective leadership that encourages total organizational involvement.
- **Quality Process** — Stressing continuous process improvement by detecting and removing defects and barriers that impact cost, quality and schedule.
- **Tools and Techniques** — The use of modern quality tools and techniques to determine root causes of defects and improve productivity.

Notes:

This FA emphasizes practices that are consistent with the principles of Total Quality Management (TQM). TQM requires integration of the quality efforts of both the program team and support elements. TQM combines quantitative methods and human involvement to improve material and services supplied to the customer. The objective of TQM is to go beyond the traditional quality control and quality assurance concepts. It is a step by step process focusing on continuous improvements. The primary responsibility for “building in” quality lies with the program members. A sound quality management process ensures that all aspects of quality are considered and acted upon by the organization. This increases the confidence of the program team members, management, suppliers and customers involved with the development, manufacture or deployment of a system.

References:

1. Harrington, H. J., pages 32-33, *The Quality/Profit Connection*, American Society for Quality Control, 1989.
2. Juran, J., *Quality Handbook*, 4th Edition, McGraw-Hill, 1988.
3. *Total Quality Management Guide*, DoD 500.51G Final Draft 2/15/90, Department of Defense, Washington, D.C.

Themes of Ensure Quality:

2.8-1 Leadership and Involvement
2.8-2 Quality Process
2.8-3 Tools and Techniques

Theme Descriptions, Typical Work Products, and Practices:

2.8-1 Leadership and Involvement

Description:

Management must demonstrate a long term commitment to sponsor and implement change even when change may be difficult or appear to have a high cost. Commitment involves visible support, adequate funding, and policies and directives. Program teams assume responsibility for their product quality, rather than relying on inspections or a quality organization to find defects.

Typical Work Products:

- Quality policy
- Funding for improvement activities
- Metrics reports
- Quality Management Plan

Specific Practices:

SP 2.8-1-1	Communicate management's role in quality improvement activities.
SP 2.8-1-2a	Assign responsibility for product quality activities and improvements to the program team.
SP 2.8-1-2b	Create an environment that encourages employee participation in identifying, reporting, and solving quality issues.

2.8-2 Quality Process

Description:

A quality process promotes continuous process improvement through detection and removal of defects and barriers that impact cost, quality, schedule, and customer satisfaction. Reductions in variability and enhancement of design robustness are key elements of defect reduction. These elements apply to both the products and the process that create the product.

Typical Work Products:

- Trouble reports
- Corrective action and lessons learned databases
- Corrective action reports

Specific Practices:

SP 2.8-2-1	Evaluate work products and system elements against requirements.
SP 2.8-2-2	Establish a process to detect the need for corrective actions to products and processes.
SP 2.8-2-3a	Evaluate processes for adherence to standards and policies throughout the system life cycle.
SP 2.8-2-3b	Perform in-progress or incremental evaluations of work products and system elements against requirements.
SP 2.8-2-4	Feed back lessons learned into processes for robustness of future designs.

2.8-3 Tools and Techniques

Description:

Modern tools and techniques exist that help determine root causes of defects and barriers to productivity and process improvements. Examples include diagramming and design of experiments.

Typical Work Products:

- Cause and effect diagrams
- Trend reports
- Pareto charts
- Fishbone diagrams
- Process maps and models
- Process simulations

Specific Practices:

SP 2.8-3-1 Use quality improvement tools in a disciplined manner to reduce defects and improve productivity.

SP 2.8-3-2 Provide readily available, just-in-time training on the use of advanced quality improvement tools.

FA 3.1 Define and Improve the Systems Engineering Process

*A Focus Area for the Systems Engineering **Environment** Category*

Description:

Define and Improve the Systems Engineering Process involves those activities needed to establish, maintain, and improve processes required to accomplish systems engineering. Ideally, the organization should strive to develop or adopt a standard systems engineering process. Then improvements to the standard systems engineering process can benefit all programs that use the process. When an organization has developed or adopted a standard systems engineering process, this standard process can be tailored for use on each program to meet program-specific needs. Each tailored process becomes an instance of the standard process that is followed by a program to meet its unique business needs.

Process management and improvement involves establishing process related criteria and improvement criteria for systems engineering activities independent of any particular program. These criteria take the form of company policies and standards for the performance of systems engineering processes. Process management establishes engineering and quality standards and guidelines for systems engineering and for the tracking and improvement of the systems engineering process.

The organization should ensure that the standard systems engineering process it develops or adopts provides complete coverage of the EIA 632 standard [1] *Processes for Engineering a System* and is consistent with both IEEE 1220-1994 [2] and EIA 632 standards for systems engineering, and should examine other system engineering standards that may be appropriate for the organization based on the industry and technologies applied.

Notes:

Prior to adoption of the standard systems engineering process developed by FA 3.1 - Define and Improve the Systems Engineering Process, the organization should use FA 1.7 - Validate System to address risks attendant to adoption. Proposed improvements to the standard systems engineering process should similarly be addressed by FA 1.7 - Validate System prior to incorporation of the improvement into the organization's standard systems engineering process. Tailoring of the standard systems engineering process to meet particular program needs becomes incorporated as part of the integrated technical program plan of FA 2.1 - Plan and Organize. FA 3.1 - Define and Improve the Systems Engineering Process is supported by an infrastructure of tools and support environment, process technology insertion, and education and training provided respectively by FA 3.4 - Manage Systems Engineering Support Environment, FA 3.3 - Manage Technology, and FA 3.2 - Manage Competency.

References:

1. EIA Standard (EIA 632), *Processes for Engineering a System*, Version 0.8, dated 6 January 1997 (or use current version). [EIA632]

2. IEEE-1220-1994, *IEEE Trial-Use Standard for Application and Management of the Systems Engineering Process*, dated February 1995 (or current version). [IEEE1220]

Themes of Define and Improve the Systems Engineering Process:

3.1-1	Systems Engineering Process Awareness
3.1-2	Establishment of a Systems Engineering Process Asset Library
3.1-3	Systems Engineering Process Development
3.1-4	Tailoring
3.1-5	Assessment of Systems Engineering Process
3.1-6	Improvements to Systems Engineering Process

Theme Descriptions, Typical Work Products, and Practices:

3.1-1 Systems Engineering Process Awareness

Description:

The program and organization examine existing systems engineering processes and selects for its use, and potential modification and development, the process(es) appropriate to meet its business goals. Among those processes examined should be the EIA 632 and IEEE-1220 standards of systems engineering as well as examining other industry-wide best practices in systems engineering.

Typical Work Products:

- Collection of systems engineering processes

Specific Practices:

SP 3.1-1-2	Establish systems engineering process goals from the organization's business goals.
SP 3.1-1-3a	Assign responsibility and provide necessary resources to plan, implement, and communicate the organization's standard systems engineering process.
SP 3.1-1-3b	Document rationale for selection and inclusion of best practices in the organization's standard systems engineering process.

3.1-2 Establishment of a Systems Engineering Process Asset Library

Description:

The organization establishes and manages a library for systems engineering process information and artifacts. A systems engineering process asset library normally contains processes, instructions for use, relationships between various entries (e.g., processes, policies, methods, guidelines, etc.), and often contains results of using the process (e.g., best practices, metrics trends, artifacts, etc.).

Typical Work Products:

- Library for Systems Engineering process information and process artifacts (collectively referred to as a *process asset library*)
- Process information and artifacts stored in systems engineering process asset library
- Instructions for use of systems engineering process asset library

Specific Practices:

SP 3.1-2-2	Establish a process library for systems engineering process assets developed and collected by the programs.

SP 3.1-2-3a Establish and assertively manage a library for systems engineering process assets developed and collected by the organization.

SP 3.1-2-3b Ensure that tailoring reports from application of the organization's standard systems engineering process to specific programs are recorded in the process library.

SP 3.1-2-3c Ensure that program results of applying the organization's standard systems engineering process are recorded in the process asset library.

3.1-3 Systems Engineering Process Development

Description:

A systems engineering processis adopted or developed for use. Generally, at the beginning of the development process, programs may be using a variety of systems engineering processes, or only part of a process. As development proceeds or matures, the organization adopts or develops a standard systems engineering process that is to be applied across all programs; each program's usage therefore becomes an instance of the standard process. The systems engineering process is developed to be well defined, with entry and exit criteria for at least all major parts of the process. The process is supplemented by an infrastructure of supporting material, such as procedures, that provide specific direction on how to accomplish process related activities. As one valuable means to develop its standard systems engineering process, the organization may look outside to other organizations to gain insight into how they perform systems engineering and to obtain best practices of systems engineering.

Typical Work Products:

- Systems engineering process(es)
- Standard systems engineering process
- Entry and exit criteria
- Procedures
- Benchmarking studies

Specific Practices:

SP 3.1-3-1 Identify existing systems engineering processes for use on programs.

SP 3.1-3-2a Establish and follow a written organizational policy (may be part of a broad-based policy) for implementing and maintaining systems engineering process(es).

SP 3.1-3-2b Describe and present the organizational policy clearly and completely to all engineering and program personnel.

SP 3.1-3-3a Plan, approve, and establish process management and improvement activities according to a formal procedure.

SP 3.1-3-3b Develop and document a standard systems engineering process for the organization based on industry standards and industry-wide best practices.

SP 3.1-3-3c Define clearly the inputs and outputs of the sub-processes that comprise the systems engineering process.

SP 3.1-3-3d Define entrance and exit criteria for each major activity in the systems engineering process.

SP 3.1-3-3e Define a set of standard methods for use with the organization's standard systems engineering process used on programs.

SP 3.1-3-3f Establish a formal process for implementing and improving Systems Engineering Activities.

SP 3.1-3-5 Integrate the systems engineering process with other engineering and enterprise processes to establish a unified product development process.

3.1-4 Tailoring

Description:

The organization's standard systems engineering process permits tailoring to meet program specific needs. A standard set of tailoring guidelines is developed to permit the standard process to be applied to various situations while maintaining the integrity of the process. The results of the program tailoring activities are reviewed by appropriate individuals (e.g., senior managers or members of an engineering process group.)

Typical Work Products:

- Tailoring guidelines
- Tailoring reports

Specific Practices:

SP 3.1-4-3a	Establish a set of tailoring guidelines for the organization's standard systems engineering process that permits the standard process to meet program-specific needs.
SP 3.1-4-3b	Tailoring reports generated by the programs are reviewed and approved by the appropriate individuals (e.g., senior managers or members of the engineering process group).

3.1-5 Assessment of Systems Engineering Process

Description:

The organization assesses the usage of systems engineering processes on programs to determine potential areas of improvement. Weaknesses and strengths of the process are identified, usually in a qualitative manner in the early stages of assessment and later in a quantitative manner as the assessment process matures. As one means to assess its processes, the organization benchmarks other organizations to assess its systems engineering process.

Typical Work Products:

- Evaluations
- Benchmarking studies

Specific Practices:

SP 3.1-5-2	Assess the program-specific systems engineering processes and determine relative strengths and weaknesses.
SP 3.1-5-3a	Assess the organization's standard systems engineering process.
SP 3.1-5-3b	Review root causes of errors or problems to determine whether changes to the systems engineering process are required to prevent future occurrences.
SP 3.1-5-3c	Use a mechanism for periodically assessing the systems engineering process.
SP 3.1-5-3d	Seek to benchmark the organization's systems engineering process against processes used by other organizations.
SP 3.1-5-3e	Determine the degree of program use of the organization's defined systems engineering process and methods.
SP 3.1-5-4a	Measure and analyze systems engineering productivity for each major process activity within the systems engineering process.
SP 3.1-5-4b	Gather and analyze data from inspections to identify areas for improvement in the systems engineering process.
SP 3.1-5-4c	Use uniform systems engineering process metrics across programs.
SP 3.1-5-4d	Use a mechanism to evaluate the utility of process metrics collected across all programs.
SP 3.1-5-5	Use a formal procedure to assure periodic management review of each program and institute changes to the systems engineering process.

3.1-6 Improvements to Systems Engineering Process

Description:

The organization improves the systems engineering process based in large part on experiential feedback from the programs. Ideally, personnel knowledgeable in program activities participate in the improvement process as well as those whose primary job function is related to process activities. Those individuals responsible for developing the systems engineering process should participate in the improvement activities. The systems engineering process and its improvement cannot effectively be managed and improved unless all affected groups are aware of, and participate in, the process activities. Improvements to the systems engineering process are communicated to all effected groups, and especially to the programs, in order that all may benefit.

Typical Work Products:

- Improvements to the systems engineering process
- Process Change Request

Specific Practices:

SP 3.1-6-2a	Perform improvement of systems engineering process(es) in use on programs in at least an informal manner.
SP 3.1-6-2b	Identify and communicate best practices within the organization to programs.
SP 3.1-6-3a	Use targeted improvements to change the organization’s systems engineering process.
SP 3.1-6-3b	Provide a mechanism for users to identify proposed improvements to the systems engineering process.
SP 3.1-6-3c	Communicate the existence and improvement of the organization’s standard systems engineering process to all affected groups and programs.

FA 3.2 Manage Competency

*A Focus Area for the Systems Engineering **Environment** Category*

Description:

The purpose of competency management is twofold: 1) to establish and nurture a learning environment which results in the needed evolution of staff knowledge and skills over time, and 2) to establish and maintain a capability to deliver the required knowledge to programs via training or other sources (e.g., consultants, contractors, subcontracting) when there is a shortfall in available resident staff knowledge. A broad and complete range of knowledge must be addressed including: systems engineering, process science, engineering disciplines (e.g., EE, ME), interdisciplinary/interpersonal, and problem domains (e.g., remote sensing, satellite communications, financial processing).

Program needs are considered when selecting from various types, sources, and opportunities for training. Effective competency development requires forecasting , planning, training media (e.g., workbooks, computer software, etc.), and a repository of competency development process data. As an organizational process, the main components of competency development include a managed competency development program, documented plans, personnel with appropriate mastery of systems engineering and other knowledge areas, and mechanisms for measuring the effectiveness of the competency development program.

Responsibility for development of competency is shared between the organization (or program) and the practitioners; both are stakeholders in this focus area. The organization is responsible for development of competency needed to meet its business goals, recognizing that achievement of these goals depends in part on its ability to recruit and retain technical staff. Individuals are responsible for identification of needed training and for participating in training to increase their ability to perform in their current assignments and to achieve career advancement goals.

Notes:

Success in competency management can be measured in terms of the availability of the required knowledge and skill to perform new and ongoing enterprise activities. Maintaining and broadening their success depends on individuals’ flexibility and motivation in acquiring knowledge. Needed skills and knowledge can be provided both by training within the organization and by timely acquisition from sources external to the organization. Acquisition from external sources may include customer resources, temporary hires, new hires, consultants, and subcontractors.

References:

Not Applicable.

Themes of Manage Competency:

3.2-1 Learning Environment
3.2-2 Competency Needs
3.2-3 Knowledge Delivery to Programs
3.2-4 Competency Assessment and Achievement

Theme Descriptions, Typical Work Products, and Practices:

3.2-1 Learning Environment

Description:

Methods for creating and nurturing a learning environment include providing management attention, training resources, tuition reimbursement programs, flexibility in work assignments, and individual recognition for competency development and training delivery. Competency development occurs on several levels: 1) university education resulting in a degree and non-degree short courses, 2) company sponsored training, and 3) on the job experiences. Each approach should be recognized and nurtured by the organization's competency development program. Effective competency development requires courseware, trainers, and environment. Courseware ranges from textbooks to domain-specific examples to current program tasks, and may be presented in printed form, audio, audio/visual, computer-based training, or other media and delivery formats. Trainers range from professors and professional systems engineering educators to mentors and peers.

Comments:

Successful training programs require an organization's commitment. They are administered in a manner that optimizes the learning process, and are repeatable, assessable, and easily changeable to meet new needs of the organization.

Typical Work Products:

- Recognition of training and training development in staff evaluations
- Awards for competency achievement
- Tuition reimbursement programs
- Competency certificates

Specific Practices:

SP 3.2-1-1	Encourage staff to continuously develop skills and knowledge.
SP 3.2-1-2a	Reward mentoring as a means of increasing staff competency.
SP 3.2-1-2b	Provide a mechanism to develop individual competency development goals consistent with both the individual's career objectives and the program's needs.
SP 3.2-1-3a	Provide job opportunity and career advancement based on competency development achievements.
SP 3.2-1-3b	Clearly state and communicate competency development opportunities and the relationship between competency development and career opportunity to all personnel within the organization.
SP 3.2-1-3c	Provide a mechanism to formally recognize competency development achievements.
SP 3.2-1-3d	Provide a mechanism for certification of competency achievement.

3.2-2 Competency Needs

Description:

Competency needs are derived from company objectives, individual skill assessments, and program needs. Needs are organized and prioritized by short term, long term, individual, program, and company. The breadth of system development is considered, including technical disciplines, processes, interpersonal skills, and problem domain areas. Mechanisms or processes are established to gather the required information on training needs.

Typical Work Products:

- Training needs assessments
- Individual training requests
- Program skill needs forecasts
- Training results evaluations

Specific Practices:

SP 3.2-2-1	Identify needed improvements in skill and knowledge throughout the organization using the programs' needs, organizational strategic plan, and existing employee skills as guidance.
SP 3.2-2-2	Base near term competency development requirements upon immediate program needs.
SP 3.2-2-3	Base long term competency development requirements upon the organization's strategic plan.

3.2-3 Knowledge Delivery to Programs

Description:

Knowledge delivery to programs involves both training of program personnel and acquisition of knowledge from outside sources. Analysis of alternatives for meeting competency needs determines the most appropriate mode of knowledge delivery. When outside sources are selected as the most effective mode, the practices of the Coordinate with Suppliers Focus Area are used. Knowledge delivery in the form of training includes preparation of training materials, conduct of training, and maintenance of training records and materials so that they are readily accessible.

Comments:

The choice of in-house training or external sourcing for the needed skills and knowledge is determined by the availability of training expertise, the program's schedule, and business goals. A primary objective of competency management should be to assure that programs have the required skills when they need them. How the skills are provided should be determined by examining alternatives according to the practices of FA 1.4 - Assess and Select.

Typical Work Products:

- Training materials and courses including, but not limited to:
    1. Systems engineering processes (requirements definition, design definition, design verification)
    2. Engineering disciplines
    3. Interpersonal/team skills
    4. Product, technology, and problem domain
    5. Consulting agreements
    6. Acquirer-supplier agreement processes
    7. Control processes
    8. Planning Processes

Specific Practices:

SP 3.2-3-1a	Train personnel to have the skills and knowledge needed to perform their assigned roles.
SP 3.2-3-1b	Maintain records of training and experience.
SP 3.2-3-1c	Provide knowledge from outside sources when in-house training or learning opportunities are unable to satisfy program needs.
SP 3.2-3-2a	Maintain training materials in an accessible repository.
SP 3.2-3-2b	Assign experienced personnel to perform training.

SP 3.2-3-2c	Involve management personnel in competency development activities, both as recipients and as participants.
SP 3.2-3-2d	Provide competency development for critical functional areas (e.g., analysis techniques specific to the organization's problem domains).
SP 3.2-3-3a	Integrate competency development opportunities, such as formal education, in-house training, and on-the-job training.
SP 3.2-3-3b	Provide cross-discipline technical management training to all disciplines, including program management.
SP 3.2-3-3c	Train managers of engineering organizations, team leaders, and engineers on the systems engineering process.
SP 3.2-3-3d	Provide training in the basic principles of systems engineering to quality management, configuration management, and other support personnel.
SP 3.2-3-3e	Provide training in a variety of forms, including formal training, on-the-job training, and just-in-time training, as required to meet program and individual needs.
SP 3.2-3-3f	Integrate tools, methods, and procedures for competency development.

3.2-4 Competency Assessment and Achievement

Description:

Competency achievement includes both assessment of results and recognition of individual and organizational achievement. Training effectiveness evaluations are conducted on both students and trainers. Evaluation information is obtained from management, programs, trainees, and training developers and fed back into training plans.

Typical Work Products:

- Awards
- Training effectiveness surveys
- Program performance assessments
- Instructor evaluation forms

Specific Practices:

SP 3.2-4-1a	Assess in-progress or completed programs to determine whether staff knowledge was adequate for performing program tasks.
SP 3.2-4-1b	Provide a mechanism for assessing the effectiveness of each training course with respect to set objectives.
SP 3.2-4-2	Require trainers to demonstrate proficiency in the topics for which they intend to train others.
SP 3.2-4-3a	Provide a mechanism to evaluate students to verify their comprehension of training materials prior to recognition.
SP 3.2-4-3b	Obtain student evaluations of how well competency development activities meet their needs.
SP 3.2-4-3c	Establish completion criteria for each training course, documented in standards or course descriptions.
SP 3.2-4-4	Provide a mechanism to evaluate alumni capability to perform the style, scope, and intensity of systems engineering that the business needs.

FA 3.3 Manage Technology

A Focus Area for the Systems Engineering ***Environment*** *Category*

Description:

Manage Technology involves identifying technologies applied to current products and processes; monitoring the progression of currently used technologies through their life cycle; identifying, selecting, evaluating, and investing in new technologies, and incorporating the appropriate technologies into the organization's products and processes to achieve competitive advantage. By maintaining an awareness of product and process technology innovations throughout the world and systematically evaluating and experimenting with them, the organization selects appropriate technologies to improve its competitiveness, and to increase both productivity and product quality. Appropriate technologies could include newly develped technologies, but could also include applying mature technologies in different applications, or maintaining current methods. Pilot efforts are performed to assess new and unproven technologies before they are introduced across the organization and, where required, investments are made to increase the maturity of the technology. With appropriate sponsorship of the organization's management, the selected technologies are incorporated into the organization's products and standard process.

Notes:

The technology management strategy should address incorporation of selected technologies, as appropriate, into the organization's standard process under FA 3.1 - Define and Improve the Systems Engineering Process. The approach to risk management defined by FA 2.5 - Manage Risk should be used to assess the risk of incorporating, or not incorporating, selected technologies into the organization's products and standard process.

References:

Not Applicable.

Themes of Manage Technology:

3.3-1 Technology Awareness, Evaluation, and Selection
3.3-2 Technology Re-use and Commercial-Off-The-Shelf Technology
3.3-3 Technology Innovations Directly Improve the Organization's Performance

Theme Descriptions, Typical Work Products, and Practices:

3.3-1 Technology Awareness, Evaluation, and Selection

Description:

The organization maintains an awareness of technologies, both improvements to existing technologies and new technologies, that are applicable to current and projected product lines. Similarly, the organization maintains an awareness of process technologies applicable to the organization's standard process. The organization selects technologies based upon first identifying and then evaluating new technologies that will provide a competitive advantage to the organization.

Typical Work Products:

- Reviews of product and process technologies
- List of candidate new technologies

- Evaluations of new technologies
- Cost benefit analysis(es)
- Return on investment
- Budgets for technology improvement

Specific Practices:

SP 3.3-1-1a	Identify technologies currently in use.
SP 3.3-1-1b	Identify new product technologies for competitive advantage.
SP 3.3-1-2	Encourage innovation within the program.
SP 3.3-1-3a	Support participation by the organization in technical consortia, societies, and collaborations.
SP 3.3-1-3b	Incorporate, as part of the organization's annual budget, participation in identification, assessment, and insertion of new technology.
SP 3.3-1-3c	Establish a mechanism for maintaining awareness and disseminating knowledge of the state-of-the-art technology.
SP 3.3-1-3d	Establish a mechanism for monitoring the life cycle of currently used technologies and use this knowledge to plan for replacement of technologies approaching obsolescence.
SP 3.3-1-3e	Perform cost/benefit analyses prior to the adoption of new technologies.

3.3-2 Technology Re-use and Commercial-Off-The-Shelf Technology

Description:

The organization examines the re-use of select technologies and available, Commercial-Off-The-Shelf (COTS) technologies to gain competitive advantage for the organization.

Typical Work Products:

- Re-use library
- COTS library

Specific Practices:

SP 3.3-2-2	Establish formal criteria for the reuse and COTS/internal development decision process.
SP 3.3-2-3a	Establish a mechanism for applying business goals to the evaluation of internal development of technologies versus those externally available.
SP 3.3-2-3b	Establish a mechanism for assessing existing designs and specifications for reuse in new applications.
SP 3.3-2-3c	Document technology improvement activities formally.

3.3-3 Technology Innovations Directly Improve the Organization's Performance

Description:

The organization manages the introduction of new technologies into the organization's processes, products, and services in order to increase value to customers and to improve the competitive position of the organization. This management should include conducting pilot(s) of proposed changes (process or product) to ensure that the intent of introducing the new technology (process or product) can actually be achieved. Risk management should be applied to the potential technology insertion. Potential risks associated with introducing the new technology should be identified, quantified, analyzed, a mitigation strategy developed and implemented, and risk monitoring conducted.

Typical Work Products:

- Implementation strategy
- Product improvements
- Process improvements

Specific Practices:

SP 3.3-3-2	Require appropriate analysis within the organization before new product or process technology insertion is allowed.
SP 3.3-3-3a	Establish a mechanism for managing and supporting the introduction of new product or process technologies.
SP 3.3-3-3b	Review the effectiveness of newly introduced technologies (product or process) to verify analysis used to justify its introduction.
SP 3.3-3-3c	Identify, discriminate, and insert product and process technology improvements.
SP 3.3-3-5	Demonstrate that the achievement of specific business goals (e.g., increased profitability, increased market share, reduced time to market) can be directly attributable to the insertion of new product or process technology.

FA 3.4 Manage Systems Engineering Support Environment

*A Focus Area for the Systems Engineering **Environment** Category*

Description:

Manage Systems Engineering Support Environment provides the environment needed to develop the product and perform the process. In this context, environment means the infrastructure (facilities and utilities) and tools that systems engineers need to perform their jobs effectively. This activity includes managing the efficiency and effectiveness of the existing environment; forecasting, planning and acquiring additional, upgraded, or new infrastructure or tools; and tailoring the existing environment for each program's needs. Responsibility for supporting the environment must be assigned. Adequate resources must also be made available to support this activity. The support environment should be managed according to a documented plan based upon the organization's goals and program requirements.

The infrastructure is the underlying office and laboratory space, furniture, fixtures, and computing and communications support for the engineers and the tool set, as well as the means of integrating individual tools to provide interoperability. Infrastructure facilities may also include machine shops, chemical processing facilities or laboratories, environmental stress facilities, and other equipment or machinery needed for prototyping or testing products or processes. Systems engineering tools span all FAs. Examples of tools include requirements analysis and management tools; modeling and simulation tools; design tools; data visualization tools; planning and scheduling tools; word processing tools; configuration management tools; and others. Technology is an enabling driver for the environment and tools. As environment and tool technologies improve, more sophisticated processes and methods can be realized, thereby improving the capability to perform work more efficiently, provided that the supporting environment is in place.

Notes:

Configuration management of the support environment is addressed in FA 2.7 - Manage Configurations. The technology needs of an organization change over time, and the efforts described in this FA should be re-executed as the needs evolve per FA 3.3 - Manage Technology, and as processes are changed per FA 3.1 - Define and Improve the Systems Engineering Process. As processes and their associated facilities and tools change, training requirements should be changed per FA 3.2 - Manage Competency.

References:

Not Applicable.

Themes of Manage Systems Engineering Support Environment:

3.4-1 Awareness of the Support Environment Needs
3.4-2 Establish the Systems Engineering Support Environment
3.4-3 Managing the Support Environment

Theme Descriptions, Typical Work Products, and Practices:

3.4-1 Awareness of the Support Environment Needs

Description:

The selection of tools and organizational infrastructure must support the organization's goals and processes while meeting the needs of systems engineers. Awareness of the current state-of-the-art or practice is a necessary element for the assessing improvement options. The support environment must be flexible enough to meet the needs of various programs.

Typical Work Products:

- Statement of needs
- Tool studies
- Vendor data

Specific Practices:

SP 3.4-1-2	Determine requirements for the support environment based on program specific needs.
SP 3.4-1-3a	Include the needs of each program as part of a documented set of requirements for the support environment.
SP 3.4-1-3b	Include the business goals of the organization in determining the documented requirements for the support environment.
SP 3.4-1-3c	Regularly review and assess external trends that might affect the support environment for potential impact.

3.4-2 Establish the Systems Engineering Support Environment

Description:

Once the organization's needs have been identified, an environment needs to be established which will satisfy the identified needs. The environment can be developed in-house or commercial tools can be acquired. The environment must support the organization's systems engineering process and be tailored to meet individual program requirements. An integrated set of tools and structures is essential to a highly effective organization.

Typical Work Products:

- User manuals and guidelines
- Environment implementation plan
- Tailoring guidelines
- Trade-off studies
- Tools

Specific Practices:

SP 3.4-2-1	Deploy a Systems Engineering Support Environment that supports program needs.
SP 3.4-2-2a	Pilot new tools prior to including them in the systems engineering support environment.
SP 3.4-2-2b	Perform cost-benefit analysis for commercial off-the-shelf versus in-house developed environments.
SP 3.4-2-3a	Establish an organizational standard system engineering support environment.
SP 3.4-2-3b	Tailor the Systems Engineering Support Environment to individual program needs.
SP 3.4-2-5	Maximize integration of tools within the environment.

3.4-3 Managing the Support Environment

Description:

Maintenance of the established support environment is critical to organizational success. Individuals must be assigned to maintain the elements of the environment to ensure that the system is available to the programs when needed. The support environment should be monitored to identify potential problems as well as to identify opportunities for improvements. End users should be surveyed to determine the adequacy of the current environment and identify potential improvements.

Typical Work Products:

- Resource budgets
- Surveys
- Lists of potential improvements
- Usage data

Specific Practices:

SP 3.4-3-1	Maintain the support environment to continuously support the program.
SP 3.4-3-2a	Assign responsibilities for maintaining the support environment.
SP 3.4-3-2b	Plan and track maintenance of the support environment.
SP 3.4-3-2c	Maintain configuration control over the support environment.
SP 3.4-3-3a	Collect data on the systems engineering support environment usage and performance.
SP 3.4-3-3b	Retire support tools or facilities which no longer support the organization’s requirements.
SP 3.4-3-3c	Upgrade or add support tools or facilities which enhance the ability to meet the organization’s requirements.
SP 3.4-3-3d	Seek periodic evaluation of the adequacy of the systems engineering support environment from users.
SP 3.4-3-4a	Base support environment management decisions on the analysis of usage and performance data.
SP 3.4-3-5	Establish goals for improvements to systems engineering processes through the use of the systems engineering environment.

Annex A (normative)

Tailoring

Tailoring refers to the application the requirements of specifications, standards, and related documents to a program/project. Tailoring is the process by which the requirements are modified to be suitable for the specific application or program.

As applied to the use of this model and the associated appraisal method, tailoring consists of 1) tailoring out (eliminating) parts of the model, such as Focus Areas (FAs) or Themes, so as to align process improvement goals and activities with organizational objectives, or 2) more typically, tailoring out higher level practices in the model, as might be done when focusing on lower capability levels.

Tailoring helps an organization focus on those parts of the model from which it can most benefit. Tailoring should be done with an awareness that it can result in significant gaps in efforts to improve or assess an organization’s capabilities.

Annex B (normative)

Glossary

For the purposes of this standard, the following definitions apply:

acquirer: An organization or individual that obtains a product**.**

acquirer-supplier agreement: An arrangement between two parties (an acquirer and a supplier) that defines the tasks to be performed and the acceptance criteria to be applied to delivered items.

activity: Any step or function performed, both mental and physical, toward achieving some objective. A task is a "formal" activity.

allocate: Assign performance requirements to a function, process, behavior, or other logical element of the system.

allocated baseline: The initially approved documentation describing a subsystem's functional, performance, interoperability, and interface requirements that are allocated from those of the system or a higher level subsystem; interface requirements with interfacing subsystems; design constraints; derived requirements (functional and performance); and verification requirements and methods to demonstrate the achievement of those requirements and constraints. Generally there is an allocated baseline for each subsystem to be developed.

appraisal method: The set of steps or procedure for conducting a systems engineering appraisal. The appraisal method consists of five phases: commitment, preparation, on-site, post-appraisal, and appraisal follow-up.

appraisal team: A team of experienced engineering professionals that are trained in the appraisal method to perform appraisal.

architecture: A high level design that provides decisions made about: the problem(s) that the product will solve, component descriptions, relationships between components, and dynamic operation description.

associated processes: Processes that enable one or more end products to be put into service, maintained in service, or disposed at the end of service.

attribute: A characteristic of an item, e.g., the item's color, size, or type. A measurable physical or abstract property of an entity.

audit: An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria.

baseline: (1) A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures. (2) A document or a set of such documents formally designated and fixed at a specific time during the life cycle of a configuration item.

benchmarking: The continuous process of measuring products, services and practices against the toughest competitors or those companies recognized as industry leaders.

capability evaluation: An independent process appraisal by a trained team of professionals.

capability level: The extent to which the organization can potentially accomplish the essential elements of systems engineering as defined in the context of SECM Focus Areas. Capability involves the attributes of people, technology, and process. SECM capability levels are an ascending scale progressing from Initial (Level 0), to Performing, to Managed, to Defined, to Measured, through Optimizing (Level 5).

Capability Maturity Model (CMM): A copyrighted term developed by the Software Engineering Institute for its maturity models.

change advocate: An individual or group who wants to achieve a change but lacks sufficient sponsorship. Contrast with change agent.

change agent: An individual or group that has sponsorship and is responsible for implementing or facilitating change. An example of a change agent is the systems engineering process group. Contrast with change advocate.

change control: (See configuration control.)

change management: The process of evaluating the impact of a requirement or design change on the system, analyzing the effects of a proposed change in terms of the system foundation architecture, performance, costs, and schedule criteria. Change management must be supported by configuration management to ensure that decisions to adopt a change in requirement, design, or implementation are reflected in system documentation, engineering drawings, or other representations of the system.

command media: An organization’s internal communications that specify how things should be done (e.g., policies, procedures, standards, work instructions).

common cause of variation: Causes of natural variation inherent in a process or system. Removing common causes of variation involves making changes to the process itself. These causes are usually minor and do not cause a process to go out of control. An example is wear and tear on equipment causing greater tolerance variation in an output, such as a drink filler at a fast-food restaurant.

completeness: As it applies to requirements, a full consideration of all implications due to higher level requirements.

compliance: Meeting the requirements of a standard or meeting specified requirements.

compliance article: An item built, constructed, or coded for the purpose of checking compliance to specified requirements.

composite results: Results which are non-specific with regard to particular individuals or programs. Typically used to refer to the findings which are presented to the senior management team during the post-appraisal phase of an appraisal.

configuration: The arrangement of the parts or elements of a work or deliverable product.

configuration baseline: The configuration information formally designated at a specific time during a system’s or subsystem’s life cycle. Configuration baselines, plus approved changes from those baselines, constitute the current configuration information.

configuration control: An element of configuration management, consisting of the evaluation coordination, approval or disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification.

configuration item: An aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management process.

configuration management: (1) A technique of applying technical and administrative direction and surveillance to (a) identify and document the functional and physical characteristics of an item, (b) control changes to those characteristics, and (c) record and report change processing and implementation status. (2) A control activity for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design and operational information throughout its life, using disciplined change management.

consistency: As it applies to requirements, an indication that upward traceability to higher level requirements is realized and that there is an absence of ambiguity, conflict, or error.

constraint: (1) A restriction, limit, or regulation. (2) A type of requirement that is not tradeable against other requirements.

continuous architecture: A capability model in which the Focus Areas are all of equal explicit priority to implement, and to which identical generic practices may be added in sets to demonstrate increasing capability. This model contrasts with the staged model that establishes sets of Focus Areas that must be performed before other sets in a priority order of increasing maturity.

corrective action: An action taken to bring expected future performance into compliance with the plan.

critical technical parameter: A parameter designated in a baseline document as critical to performance.

customer: A purchaser or user of end products.

customer expectations: What a customer expects to receive from a supplier after the supplier has committed to product or system requirements.

customer requirements: The set of essential customer needs, expressed as what the customer wants and why. The requirements comprehend the problem that the customer wants to solve. Statements of fact and assumptions that define the expectations of the system in terms of mission or objectives, environments, constraints, and measures of effectiveness. These requirements are defined from a validated needs statement (Mission Needs Statement), from acquisition and program decision documentation, and from mission analyses of each of the primary system life cycle functions.

customer satisfaction: The results of delivering a product that meets customer requirements.

data: The various forms of documentation required to support a program in all of its areas. Data may take any form (e.g., printed or drawn on various materials, electronic media, or photographs). Data may be deliverable to a customer or non-deliverable for internal use only.

data management: Administrative control of program data, both deliverable and non-deliverable. Administrative control involves such items as identification, interpretation of requirements, planning, scheduling, control, archiving and retrieval of program data.

defect review: A review of a work product, interim or deliverable, that occurs prior to the release of the work product to the next process step. The review involves the creator of the product and subject matter peers, including an outside reviewer for objectivity, who identify defects in the product that would make it unsuitable for use in the next work process, and also develop a common vision of the work product. It is a form of both static testing of the work product earlier than its production and a communication mechanism. See also peer review.

defined process: A repeatable process that has clearly stated inputs, entry criteria, activities, roles, measures, verification steps, outputs, and exit criteria. A defined process is typically defined at the organizational level or tailored from the organization's set of standard processes. Exceptions are documented, reviewed, and approved. A defined process is well characterized and understood, and is described in terms of roles, standards, tools, and methods. A defined process is described formally in an organization for use by its managers and practitioners. This description may be contained, for example, in a document or a process asset library. The defined process is what the organization's members are supposed to do. A defined process is developed by tailoring the organizational process to fit the specific characteristics of its intended use. (See also organizational process.)

deliverable: An item agreed to be delivered to an acquirer as specified in an acquirer-supplier agreement. This item can be a document, a hardware item, a software item, or any type of work product.

delivered products: Those work products that the customer receives. These may also include specifications, interim documents, and prototypes, in addition to the final end product.

derived requirement: (1) A requirement that is further refined from a primary source requirement or a higher level derived requirement. (2) A requirement that results from choosing a specific implementation for a system element.

design: The set of decisions about a product that results in a common vision of what need it addresses, and how it addresses or satisfies that need. Typically, a design includes an operational concept (how users are expected or intended to use the product), components and their relationships, and sometimes decisions about the processes that will produce, deploy, and support it.

detailed design baseline: The Design Solution resulting from Detailed Design phase activities (e.g., detail drawings, detail specifications, other Design Solution descriptions placed under configuration control).

> Note: This baseline is used to build or construct the compliance articles to be used in the System Verification and Validation Process activities and tasks.

developer: An organization that performs development activities.

development: Activities for applying technology to the problems of society.

development baseline: An agreed upon description of the attributes of a building block that serves as a basis for defining change and qualifying the end products and associated processes, recorded at a point in time.

development life cycle: A progression from inception to completion of development of a system.

development phase: A group of defined activities in a unit of a development life cycle.

document: A collection of data, regardless of the medium on which it is recorded, that generally has permanence and can be read by humans or machines.

domain: A subject area that provides some benefit(s) to practitioners to perform. Examples include software or systems engineering, human resources, marketing, finance, facilities construction, renovations and demolition, catering, hospital operating or emergency rooms, and legal or professional advice.

effectiveness: A measure of the performance of an activity. SECM characterizes effectiveness as marginal, adequate, significant, measurably significant and optimal. These are defined as follows:

> **marginal effectiveness:** Effort is being expended but it is not clear that the benefit received for the effort invested is worth the cost of the effort. The effort could be removed without causing significant impact to the program or organization.

> **adequate effectiveness:** Effort is being expended and the activities provide reasonable benefit to the program or organization.
>
> **significant effectiveness:** Effort being expended is obviously beneficial to the program or organization.
>
> **measurably significant effectiveness:** Effort being expended and the benefit are measured and found to be significant to the program or organization.
>
> **optimal effectiveness:** Effort expended is providing maximum benefit for the amount of effort, i.e., more effort results in a diminishing return to the program or organization.

effectiveness assessment: An analysis of how well a product associated with a Design Solution will perform or operate given anticipated usage.

effective process: A series of actions that, when properly performed, produce the intended result (e.g., the desired state change in an object). Given several equally effective processes, their relative efficiency, in terms of resource consumption, can be empirically determined. An effective process can be characterized as practiced, documented, enforced, trained, measured, and able to improve.

empowerment: The alignment of decision making and its authority, consequences, information, and capability to perform with the goals to be achieved.

end system: The topmost building block in a hierarchy of building blocks. The system that is self contained in terms of its use and operation.

engineering plan: The plan for guidance and control of the technical efforts on a program. The engineering plan reflects an integrated technical effort responsible for product development that balances all factors associated with meeting system life cycle requirements.

enterprise: The legal entity within which an organization resides. A unit within a legal entity or spanning several entities, within which one or more programs are managed as a whole. All programs within an enterprise, at the top of the reporting structure, share a common manager and common policies. (See also organization.)

environment: (1) The natural conditions (weather, climate, ocean conditions, terrain, vegetation, dust, etc.) and induced conditions (electromagnetic interference, heat, vibration, etc.) that constrain the Design Solutions for end products and their enabling products. (2) External factors affecting an organization or program. (3) External factors affecting development tools, methods, or processes.

error prevention analysis: A process that is typically conducted by a working group of engineering professionals who developed the documentation/product in question. It is an objective assessment of each error, its potential cause, and the steps to be taken to prevent it. While placing blame is to be avoided, such questions as mistakes, adequacy of education and training, tools capability, and support effectiveness are appropriate areas for analysis.

executive: A management role at a higher level in an organization where the primary focus is the long-term vitality of the organization, rather than short-term program and contractual concerns and pressures.

exit criteria: Specific accomplishments or conditions that must be satisfactorily demonstrated before an effort can progress further in the current life cycle phase or transition to the next phase.

facilitator: Expert in systems engineering process appraisal responsible for guiding a systems engineering process appraisal team members through a systems engineering process appraisal activity.

findings: The conclusions of an appraisal, evaluation, audit, or review that identify the most important issues, problems, or opportunities within the area of investigation.

first line manager: The first level of management responsible for reviewing performance and salary; the manager directly responsible for supervising the working level staff. First line managers are responsible for managing technical supervisors and professionals.

formalization: Documentation, training, mentoring, deployment of improvements, and other mechanisms that seek to make a work process more formal, consistent, or standardized across the operating unit. (See also institutionalization.)

formal method: A technique for expressing requirements in a manner that allows the requirements to be studied mathematically. Formal methods allow sets of requirements to be examined for completeness, consistency, and equivalency to another requirement set. Formal methods result in formal specifications.

formal procedure: A documented series of steps with guidelines for use.

formal reviews: A review that is conducted in accordance with approved and established standards.

function: A task, action, or activity performed to achieve a desired outcome.

functional analysis: Examination of a defined function to identify all the sub-functions necessary to the accomplishment of that function; identification of functional relationships and interfaces (internal and external) and capturing these in a functional architecture; and flow down of upper level performance requirements and assignment of these requirements to lower level sub-functions.

functional architecture: The hierarchical arrangement of functions, their internal and external (external to the aggregation itself) functional interfaces and external physical interfaces, their respective functional and performance requirements, and design constraints.

functional baseline: The initially approved documentation describing a system's or configuration item's functional performance, interoperability, and interface requirements and the verification required to demonstrate the achievement of those specified requirements.

functional requirement: A requirement that defines a function of the system under development.

generic practice: A practice which is added to the specific practices of a Focus Area to create a higher capability process. A generic practice should be generally applicable to any Focus Area.

implementation: In product life cycle context, the process of translating a product design into the product.

implementation model: A description of not only what, but how people, methods, materiel, and equipment are applied to produce an output or outcome.

implemented process: The process that members of programs in the organization actually do. Same as "performed process."

informal review: A review that is conducted in an ad-hoc manner.

infrastructure: All the systematic elements needed to sustain an initiative or effort. These are, at a minimum: people's skills and knowledge or the training to transition them into use, methods or techniques, materiel (inputs), facilities, and tools which may or may not be automated.

inspection: The examination (review) of a product and its associated documentation to determine whether or not it conforms to requirements.

institutionalization: The building and reinforcement 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.

integrated database: A database that contains work products and outcomes from implementation of the processes for engineering a system.

Notes:

1. This database provides the information needed by the multidisciplinary teams and management to efficiently and effectively accomplish their assigned tasks. It typically contains the stakeholder needs, operational concept, operational requirements, and system requirements.

2. This database will contain (a) the current configuration of a system, (b) the current configuration baselines, and (c) all analysis and test results leading to decisions that affect the systems configuration or that are used in verifying the planned life cycle events.

integrated product development (IPD): A systematic approach to product (or service) development that achieves a timely collaboration of necessary disciplines throughout the product life cycle to better satisfy customer needs.

integrated product team (IPT): A team that includes stakeholders to product success and have accountability for producing it, including later life cycle disciplines. It may also include non-member interfaces to less critical stakeholders, and affected groups that are not users (such as community representatives downwind of a chemical plant). Deploying IPTs does not necessarily constitute performing IPD.

integration: The merger or combining of two or more elements (e.g., components, parts, or configuration items) into a functioning and higher level element with the functional and physical interfaces satisfied.

life cycle: The scope of systems or product evolution beginning with the identification of a perceived customer need, addressing development, test, manufacturing, operation, support and training activities, and continuing through various upgrades or evaluations until the product disposal.

maintenance: The process of modifying a product or component after delivery to correct faults, adapt to a changed environment, improve performance or other attributes, or perform line and depot maintenance of hardware components. That is, it includes maintenance that may be corrective, adaptive, or perfective.

measure: A reference standard or sample used for quantitative comparison of properties.

measurement: Data collected on a process, task, or activity, or the action of collecting said data. Measurements are used to synthesize metrics.

mechanism: A means or technique whereby the performance of a task, procedure, or process is assured. The mechanism may involve several organizational elements, and its documentation may include some combination of function statements, operating plans, position descriptions, and formal procedures. The documentation defines what should be performed, how it should be performed, and who is accountable for the results.

method: A specific set of rules, techniques, or guidelines for carrying out a process and its activities. Thus a method serves to organize and discipline the overall process of developing and evolving systems.

metrics: A synthesis of multiple measurements for the purpose of defining a process characteristic. (See also planning and control metric, and technical performance measures.)

model: A simplified representation of some aspect of the real world.

multidisciplinary teamwork: The cooperative application of all appropriate disciplines by people functioning as a team to achieve solutions that balance the contributions of the disciplines in an effective manner.

need: A user related capability shortfall (such as those documented in a need statement, field deficiency report, or engineering change order), or an opportunity to satisfy a new market or capability requirement because of a new technology application or breakthrough, or to reduce costs. Needs may also relate to providing a desired service (e.g., system disposal).

normative: Relating to, or prescribing, a norm or standard.

operational scenario: A sequence of events expected during operation of system products. Includes the environmental conditions and usage rates as well as expected stimuli (inputs) and responses (outputs).

operations product: One or more end products that, together, perform the operations function of a system.

opinion leader: An engineering professional highly respected by his/her peers and whose opinion can significantly influence his/her peers.

organization: A unit within an enterprise, the whole enterprise or other entity (e.g., government agency or branch of service), within which many programs are managed as a whole. All programs within an organization typically share common policies at the top of the reporting structure. An organization may consist of collocated or geographically distributed programs and supporting infrastructures. The term “organization” is used to connote: an infrastructure that supports common strategic, business, and process-related functions. The infrastructure exists and must be maintained for the business to be effective in producing, delivering, supporting, and marketing its products. The organization is the “permanent” part of the business entity that encompasses what is commonly termed the “corporate memory.”

organizational process : A process described at the organizational level for use by programs in the organization. It may be a family of processes in order to capture the different classes of processes that frequently occur in organizations. It is intended that the organizational process be tailored into a well-defined process to meet the needs of specific programs.

organization process maturity: The extent to which an organization has explicitly and consistently deployed processes that are documented, managed, measured, controlled, and continually improved. Organization process maturity may be measured via a process appraisal.

organizational unit: A single, defined organizational component (e.g., a department, section, project or program).

partition: The assignment of logical arrangements of requirements to potential end products, manual operations, or associated processes.

peer review: A review of a work product, following defined procedures, by peers of the product’s producer for the purpose of identifying defects and improvements.

performance: A quantitative measure characterizing a physical or functional attribute relating to the execution of a mission or function. Performance attributes include quantity (how many or how much), quality (how well), coverage (how much area, how far), timeliness (how responsive, how frequent), and readiness (availability, mean time between failures). Performance is an attribute for all products, processes, and their associated personnel, including those for development, production, verification, deployment, operations, support, training, and disposal. Thus, supportability parameters, manufacturing process variability, reliability, and so forth, are performance measures.

performance requirement: How well the system products must perform a function along with the conditions under which the function is performed.

performed process: What the members of the organization actually do. Also referred to as the "implemented process."

plan: A documented series of tasks required to meet an objective, typically including the associated schedule, budget, resources, organizational description and work breakdown structure.

planning and control metric: A combination of measurements used to provide periodic assessment of the health and status of a program throughout its life cycle. These metrics are used to detect the presence of adverse trends early enough so that corrective actions may be taken. Examples of planning and control metrics include: schedule performance vs. plan; cost performance vs. plan; staffing level actuals vs. planned.

policy: A guiding principle designed to influence or determine decisions, actions, or other matters.

practitioner: Anyone who participates in the systems engineering process. Practitioners may be individuals responsible for accomplishing systems engineering tasks, individuals that use the products of systems engineering, or individuals that provide resources to enable systems engineering process implementation.

procedure: A description of a course of action to perform a given task.

process: A set of interrelated activities that, together, transform inputs into outputs.

process asset library: A collection of process assets, maintained by an organization, for use by programs in developing, tailoring, maintaining, and implementing their defined processes. This collection exists within a defined architecture that gives structure to the example processes, process fragments, process-related documentation, process architectures, process tailoring rules and tools, and process measurements.

process assets: A collection of items, maintained by an organization, for use by programs in developing, tailoring, maintaining, and implementing their processes. These process assets typically include:

- the organization standard processes,
- descriptions of the life cycle models approved for use on programs,
- the guidelines and criteria for tailoring the organization's standard processes,
- the organization's measurement database, and
- a library of process-related documentation.

Any item that the organization considers useful in performing the activities of process definition and maintenance could be included as a process asset.

process capability: The range of expected results that can be achieved by following a process.

process capability baseline: A documented characterization that defines the process capability of a specific process. A process capability baseline is typically established at an organizational level.

process database: A repository into which all process data are entered. It is a centralized resource managed by the process group. Centralized control of this database ensures that the process data from all programs are permanently retained and protected.

process effectiveness: Process effectiveness is focused on the results of performing a process. These results are based on an organizational context that includes many attributes in addition to its process-related attributes. The aspects of effectiveness measured by the SECM are the approach to the process, and its deployment. The approach is characterized by practices that are expected to produce increased benefit when performed appropriately in their organizational context. The deployment is characterized by goals that reflect the end state that performing a group of practices would be expected to exhibit.

process group: A group of specialists that facilitates the definition, maintenance, and improvement of the processes used by the organization.

process management: The set of activities, methods, and tools applied to the definition, implementation, monitoring and improvement 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 program or organization takes into account both product and process-related factors in planning, performance, evaluation, monitoring, and corrective action.

process maturity: The extent to which a process is explicitly documented, managed, measured, controlled, and continually improved.

process metrics: Quantitative data used for assessing the effectiveness of the process and identifying corrective actions to be taken.

process performance baseline: A documented characterization of a process that is used for comparing actual process performance against the expected process performance. A process performance baseline is typically established at the program level, as the process is implemented. (See process capability baseline for contrast.)

producibility: A measure of the relative ease of manufacturing a product.

product: Any output or observable outcome of an activity or process, including those from services, intended for delivery to a customer or end user. This includes: (1) An item that is the goal of the engineering of a system. (2) A constituent part of the system. (3) Goods and services.

> Note: An element performs one or more of the eight primary system functions: development, manufacturing, verification, deployment, operations, training, support, disposal. The product will consist of one or more of the system element types: hardware, software, facilities, data, materiel, personnel, services, and techniques.

product measures: Measurable attributes of a product, such as size or number of defects, that generally do not vary over time (i.e., the product measure can be measured at any time).

profile: A representation (e.g., chart, spreadsheet, etc.) of the actual versus planned status of an item, action or task over time.

program: A set of tasks that are oriented towards meeting specific, defined objectives and accomplished by a group of individuals. The set of tasks are generally complex in nature and are performed within a definable time span (time between start and completion can often span numerous years) according to a planned schedule that has intermediate milestones.

program manager: The individual ultimately responsible for accomplishing the tasking and meeting the objectives, schedule and fiscal constraints of the program. Everyone working on the program is tasked by the program manager, either directly or by delegated authority through subordinate managers.

project: A development effort consisting of both technical and management activities for the purpose of engineering a system.

project leader: A designated individual with responsibility and authority over a project.

project manager: The role with responsibility for an entire project. The project manager directs, controls, administers and regulates a project.

project representatives: The engineering professionals representing a project to be assessed - typically the project leader and any optional support from one or two of the project technical professionals.

project's defined process: The operational definition of the process as used by a specific project. Well characterized and understood, it is described in terms of roles, standards, procedures, tools, and methods. It is developed by tailoring the organizational process to fit the specific characteristics and objectives of the project.

prototype: A model (physical, electronic, digital, analytical, etc.) of a product built or constructed for the purpose of: (a) assessing the feasibility of a new or unfamiliar technology, (b) assessing or mitigating technical risk, (c) validating requirements, (d) demonstrating critical features, (e) qualifying a product, (f) qualifying a process, (g) characterizing performance or product features, or (h) elucidating physical principles.

qualification: (1) The formal act of determining that the Design Solution satisfies its specified requirements of the development baseline. (2) The formal act of determining that the requirements of the associated processes will be satisfied in time to meet the needs of the related end products and that the methods, tools and environment of each process will perform their intended functions.

qualify: (1) Determine formally that a design solution satisfies its specified requirements of the development baseline. (2) Declare suitability and readiness for production, deployment, training, operations, support, and disposition.

quality: The attribute of a product, by which satisfaction of requirements is measured.

Quality Function Deployment: A formal method for establishing and prioritizing customer needs and organizational capabilities to meet them, then tracing the deployment of processes and improved capabilities to deliver them through production and distribution of the product or service.

quality goals: Specific objectives, which if met, provide a level of confidence that the quality of a product is satisfactory.

reference model: A model that is used as benchmark for measuring some attribute.

regression testing: The testing required to determine that a change to a system component has not adversely affected functionality, reliability or performance.

reliability: The probability that a system or component will perform its required functions under stated conditions for a specified period of time.

repeatable process: A set of activities performed, in an essentially identical manner on many programs, to achieve a given purpose (process) that is:

- guided by organizational policies,
- documented and planned,
- allocated adequate resources (including funding, people, and tools),
- staffed with responsibilities assigned,
- implemented by trained individuals,
- measured,
- tracked with appropriate corrective actions, and
- reviewed by appropriate levels of management.

requirement: Something that governs that a product will have a given characteristic or achieve a given purpose, including what, how well, and under what conditions.

requirements analysis: The determination of system specific performance and functional characteristics based on analyses of: customer needs, requirements, and objectives; mission/operations; projected utilization environments for people, products, and processes; constraints; and measures of effectiveness. The bridge between customer requirements and system specific requirements from which solutions can be generated for the primary system functions or characteristics.

requirements traceability: The evidence of an association between a requirement and its parent requirement or between a requirement and its implementation.

resourcing, resourced: Having to do with identifying, acquiring, and applying or deploying resources needed by a process or activity.

responsibility: A duty to provide, or contribute in a particular way to provide, a specified output or outcome, and the accountability to provide those expected results.

Return on Investment (ROI): The ratio of earnings from output (product) to investment and production costs, which determines whether an organization benefits from performing an action to produce something.

review: (See formal review, informal review and peer review.)

review data: The data that is gathered from requirements or design reviews. These data are of two types. The first, concerning the review process, typically includes preparation time, errors identified during preparation (by category), hours per error found in preparation, review time, number of requirements or design statements reviewed, number of requirements or design statements reviewed per hour, and errors found per review man-hour (by category). The second type, product data from the review, typically includes errors found per requirement or design statement, action items identified from each review, action items closed for each review, items needing re-review, re-reviews conducted.

review efficiency: The percentage of errors found through the review process. It is typically stated as a percentage and is calculated by dividing the total errors found during review by the total errors found by both review and test through the completion of system integration test.

risk: (1) A measure combining the uncertainty of reaching a goal with the consequences of failing to reach the goal. (2) The probability of suffering injury or loss.

> Note: Risk depends on the probability of occurrence and the consequences of occurrence. Risk is assessed for program, product, and process aspects of the system. This includes the adverse consequences of process variability. The sources of risk include technical (e.g., feasibility, operability, producibility, testability, and systems effectiveness); cost (e.g., estimates, goals); schedule (e.g., technology/material availability, technical achievements, milestones); and programmatic (e.g., resources).

risk management: An organized process for identifying and assessing risks and implementing means to maintain them at an acceptable level.

self-appraisal: A systems engineering appraisal for which: (a) all or most of the appraisal team is from the appraised organization (not necessarily the appraised site) and (b) the appraised organization has primary responsibility for facilitation and planning.

simulation: A model of a product or its component(s) in operation. Simulations may be computer-based analogies to a product such as video games, or a practice role-play of an operation with or without tools (a simulation of a method or technique as in training classes).

special causes of variation: Special causes of variation are assignable to people, places, materials, events, and so forth. They are causes of variation that are not attributable to, or inherent in, the process itself, although they may be attributable to some aspect of its execution.

specific practice: A practice contained in a Focus Area that describes an essential activity to accomplish the purpose of the Focus Area, is not generic to all Focus Areas, and contains a description, typical work products in which one might find evidence the practice is performed and how well it is performed, and notes, including explanations, context, references for "how to," etc.

specification: A document that clearly and accurately describes requirements and other characteristics for a product and the procedures to be used to determine that the product satisfies these requirements.

> Note: Two types of specifications are defined in this Standard: performance and detail. Three states of specifications are defined in this Standard: conceptual, initial, and definitized.

sponsor: The senior site executive who has committed to or requested the systems engineering appraisal. This person typically has control of financial and other resources for the systems engineering organization.

staged architecture: A capability model in which sets of Focus Areas are defined as maturity levels that must be completed before other sets are attempted. This explicit prioritization of Focus Areas establishes an infrastructure or other dependencies perceived to exist by the model developers or community. Contrast with the continuous architecture in which all Focus Areas have explicit equal priority, and increasing process capability is within those Focus Areas chosen by the organization to improve.

stakeholder: An individual or organization interested in the success of a product or system. Examples of stakeholder include customers, developers, engineering, management, manufacturing, users, etc.

stakeholder requirement: Represents the needs and expectations of a stakeholder.

standard: A document that establishes engineering and technical requirements for processes, procedures, practices and methods that have been decreed by authority or adopted by consensus.

statement of work: A description of all work required to complete a program or project. A contractual document which defines the work effort required from contractors, suppliers, or customer support activities.

statistical process control: Establishment and use of statistical process control chart(s) (any or all of several types) to measure and identify out-of-control conditions in a process and take action to return the process to an in-control state.

statistical quality control: Statistically based analysis of a process, measurements of process performance including identification and elimination of common and special causes of variations in the process, and the designation and implementation of improvements to change the process performance in the direction of improved customer value or operational efficiency.

subcontractor: An individual, partnership, organization, enterprise, or association that contracts with an organization to perform, or help to perform, some part(s) of the contracting organization's product life cycle: management, requirements, design, development, production, delivery, support, and/or disposal of a product or its by-products.

subsystem: A grouping of items that will perform a logical set of functions within a particular end product.

supplier: Provides a product or component to an acquirer.

sustainability: The product attribute of being maintainable in an operable condition whether the product is in use or not by customers or end users. Includes supply aspects if the product consumes materiel.

synthesis: The translation of input requirements (including performance, function, and interface) into possible solutions (resources and techniques) satisfying those inputs. Defines a physical architecture of people, product, and process solutions for a logical grouping of requirements (performance, function, and interface) and then designs those solutions.

system: The aggregation of end products and enabling products that achieves a given purpose.

system architecture: A logical, physical structure that specifies interfaces and services provided by the system components necessary to accomplish system functionality.

system component: A basic part of a system. System components may be personnel, hardware, software, facilities, data, materiel, services, and or techniques which satisfy one or more requirements in the lowest levels of the functional architecture. System components may be subsystems and/or configuration items.

System Design Process: A process for converting stakeholder requirements into Design Solutions.

systems engineering: An inter-disciplinary approach and means to enable the realization of successful systems.

systems engineering process group (SEPG): A group focused on improving the systems engineering process used by an organization. The SEPG defines and documents the systems engineering process, establishes and defines process metrics, supports program data gathering, assists programs in analyzing data, and advises management of areas requiring further attention.

system requirements: Characteristics of a system that identify the accomplishment levels needed to achieve specific objectives for a given set of conditions.

system technical requirement: A deconflicted set of stakeholder requirements stated in technical terms.

tailoring: The process by which the requirements of specifications, standards, and related documents are modified to be suitable for a specific application or program.

task: Well-defined unit of work in a process that provides a visible checkpoint into the status of the product of the process. Tasks have readiness (entry) criteria and completion (exit) criteria. Activities are informal tasks or steps within a task.

team: A group of individuals that are mutually responsible for achieving goals. Teams usually are limited to five to nine members, but may include interface representatives for stakeholders in the team's goals, and appear much larger, as in teams of team representatives, for example.

technical effort: Activity that influences system performance by defining, designing, or executing a task, requirement or procedure. All the activities required to implement and execute the systems engineering process are technical efforts.

technical objectives: Technical objectives or goals guide the development effort by providing "target" values for item characteristics. These can include cost, schedule, and performance attributes deemed important. Technical objectives are not specification requirements.

technical performance measurement (TPM): The technique of predicting the future value of key technical parameters of the end system based on current assessments of systems that make up that end system.

Notes:

1. Involves the continuing verification of the degree of anticipated and actual achievement for technical parameters. Confirms progress and identifies variances that might jeopardize meeting an end system requirement. Assessed values falling outside established tolerances indicate a need for evaluation and corrective action.

2. Key characteristics of TPM are:

   (a) Achievement-to-Date presents achieved value of the technical parameter based on estimates or actual measurement;

   (b) Current Estimate is the value of the technical parameter predicted to be achieved by the end of the technical effort with remaining resources (including schedule and budget);

   (c) Technical Milestone is a point where TPM evaluation is accomplished or reported;

(d) Planned Value Profile is the time-phased achievement projected for the technical parameter from the beginning of the development or as replanned as a result of a corrective projection;

(e) Tolerance Band is an envelope containing the Planned Value Profile and indicating the allowed variation and projected estimation error;

(f) Objective is the goal or desired value at the end of the technical effort;

(g) Threshold is the limiting acceptable value that, if not met, would jeopardize the program;

(h) Variation is the difference between the Planned Value and the Achievement-to-Date value.

technical performance parameter: Technical Performance Parameters are a selected subset of the system's performance parameters used as the technical measures tracked in TPM. Technical Performance Parameters can be:

- Specification Requirements,
- Performance parameters such as measures of effectiveness and other key decision metrics used to guide and control progressive development, and
- Design to cost requirements or goals.

technology: The tools, equipment, techniques, and methods that can be applied by people to accomplish a particular result.

test: An activity in which a system, product, or a component is used under specified conditions, the results are observed or recorded, and an evaluation is made as to whether it adequately meets some or all of its requirements.

Total Quality Management (TQM): An industry term for a variety of improvement methods and techniques that generally focus on process improvement using measurement.

tool: Typically, a computer program used to help automate tasks associated with the definition, synthesis, test, analysis, or maintenance of models, designs, and documentation associated with systems components.

traceability: (1) The ability to trace the heritage and lineage of a requirement. (2) The ability to show upward compliance of derived requirements with higher level parent requirements and downward completeness of derived requirements from higher level parent requirements.

trade study: An objective evaluation of alternative requirements, architectures, design approaches, or solutions using identical ground rules and criteria.

training: Instruction and applied exercises for the attainment and retention of skills and knowledge required to accomplish necessary tasks.

transitioned: Indicates that a tool, technique, method, process, or product has been placed into use in an operational environment where it was not used before. The implication is that personnel or users were trained, and the method or product is actually the way that they do work now or the object that they use now.

users: Individual or organization that uses, applies, or operates an end or enabling product.

user requirements baseline: A configuration baseline of user requirements. This baseline is necessary to systems engineering domains in which there does not exist a single, clearly discernible, user or customer.

validation: (1) An activity that ensures that a set of technical requirements is consistent and complete with respect to parent requirements. (2) An activity that ensures that an end product stakeholder's true needs and expectations are met.

value: A measure of the desirability of the products of an activity. SECM characterizes value as marginal, adequate, significant, measurably significant and optimal. These are defined as follows:

> **marginal value:** Products are generated by the activity, but it is not clear that the products are of use to those for whom they are intended. The products could be removed without causing significant impact to the program or organization.
>
> **adequate value:** Products generated by the activity provide reasonable benefit to those that use them. Products providing adequate value are generally used by those for whom they are intended.
>
> **significant value:** Products generated by the activity are obviously beneficial by those that use them. Products of significant value are avidly sought out and used by those for whom they are intended.
>
> **measurably significant value:** The benefits of each product generated by the activity are measured and found to be of significant value to the program or organization.
>
> **optimal value:** Value of the products generated by activity are of maximum utility to the program or organization.

variation: Difference between the planned value of the technical parameter and the achievement-to-date value derived from analysis, test, or demonstration.

verification: An activity that ensures that the selected design solution satisfies the detailed technical requirements.

well-defined process: A documented, consistent, and complete process with entry criteria, inputs, task descriptions, verification descriptions and criteria, outputs, and exit criteria.

work breakdown structure (WBS): A product-oriented family tree composed of hardware, software, services, data, and facilities which results from systems engineering efforts and which completely defines the program. Displays and defines the product(s) to be developed or produced, and relates the elements of work to be accomplished to each other and to the end product.

work product: Anything produced by a process. This includes files, documents, components, work-in-progress, specifications, invoices and so forth, generated during process performance, not just the product delivered to the process customer or user.

Annex C (informative)

Requirements

This document is based on the requirements expressed in the EIA Systems Engineeing Capability Model Working Group Charter:

- Establish a single SE Capability Model and appraisal method by merging the INCOSE & EPIC Models
- Use consensus for working group decision-making
- Comprise the working group of 2 EPIC, 2 INCOSE, and 3 (preferably 4, including the chair) EIA representatives to represent their respective interests
- Submit the merged product(s) as a proposed standard
- Provide complete coverage of EIA632

Annex D (informative)

Acronyms

AA	Advanced Attribute
ANSI	American National Standards Institute
BIT	Built-In-Test
CAWG	Capability Assessment Working Group
CDRL	Contract Data Requirements List
CMM	Capability Maturity Model
COTS	Commercial-Off-The-Shelf
CR	Change Request
DMP	Data Management Plan
DSMC	Defense Systems Management College
EIA	Electronic Industries Association
EPIC	Enterprise Process Improvement Collaboration
FA	Focus Area
GA	Generic Attribute
GC	Generic Characteristic
GP	Generic Practice
IEEE	Institute of Electrical and Electronic Engineers
IMP	Integrated Master Plan
IMS	Integrated Master Schedule
INCOSE	International Council of Systems Engineering
ISO	International Organization for Standards
NDI	Non-Development Item
PA	Process Area
PDP	Product Development Process

QFD	Quality Function Deployment
SECAM	Systems Engineering Capability Assessment Model
SECM	Systems Engineering Capability Model
SE-CMM	Systems Engineering - Capability Maturity Model
SP	Specific Practice
SPICE	Software Process Improvement Capability Determination
TC	Technical Category
TPM	Technical Performance Measurement
TPP	Technical Performance Parameter
TQM	Total Quality Management
WBS	Work Breakdown Structure

Annex E (informative)

Previous Acknowledgments

This document was generated primarily from information contained within the INCOSE Systems Engineering Capability Assessment Model (SECAM), Version 1.50, and the EPIC Capability Maturity Model for Systems Engineering (SE-CMM), Version 1.1. These two source documents, in turn, were built upon previous version(s) of these documents.

The contributors to the versions of the INCOSE SECAM were as follows:

INCOSE SECAM, Version 1.00, dated February 1994:

Don Crocker	Grumman
Doug Low	Hughes Aircraft Company, Electro-Optical Systems
Bill Mackey	Computer Sciences Corporation
Bill Money	Loral Command and Control Systems
Hal Pierson	Software Productivity Consortium
Al Reichner	Loral Space and Range Systems

INCOSE SECAM, Version 1.10, dated July 1994:

Blake Andrews	Rockwell Collins Air Transport Division
Roger Bate	Software Engineering Institute
Jerry Burleson	Loral Command and Control Systems
Don Crocker	Grumman
Kevin Forsberg	Center for Systems Management
Suzanne Garcia	Software Engineering Institute
Bob Jones	Loral Federal Systems
Dorothy Kuhn	Texas Instruments
Bill Mackey	Computer Sciences Corporation
Don Marquet	MMC Astronautics
Bill Money	Loral Command and Control Systems
Richard Pariseau	Naval Air Warfare Center, Aircraft Division
Tom Paruer	AAI
Hal Pierson	Software Productivity Consortium
Dave Preklas	GTE
Art Pyster	Software Productivity Consortium
Al Reichner	Loral Space and Range Systems
Jim Rogers	Hewlett Packard
Mary Simpson	Battelle - Pacific NW Labs
George Stern	TASC
Don Wick	Hewlett Packard
Rich Widmann	Hughes Aircraft Company, Electro-Optical Systems

INCOSE SECAM, Version 1.20, dated November 1994:

Blake Andrews	Rockwell Collins Air Transport Division
LeRoy Botten	Computer Sciences Corporation
Vic Church	Computer Sciences Corporation
Dennis Crehan	Loral Aerosys
Raymond Granata	NASA, Goddard
Joe Ludford	Computer Sciences Corporation
Bill Mackey	Computer Sciences Corporation
Stephan Mayer	Allied Signal
Bill Oran	Allied Signal
Steven Senz	GTE
Bruce Shelton	Systems Management Development Corp.
Jesse Silvers	Computer Sciences Corporation
Antonio Vallone	Computer Sciences Corporation
George Van Nostrand	Computer Sciences Corporation
Roger Werking	Computer Sciences Corporation
Rich Widmann	Hughes Aircraft Company, Electro-Optical Systems

INCOSE SECAM, Version 1.30/1.31, dated April 1995:

Blake Andrews	Rockwell Collins Air Transport Division
LeRoy Botten	Computer Sciences Corporation
Doug Low	Hughes Aircraft Company, Electro-Optical Systems
Bill Mackey	Computer Sciences Corporation
Richard Pariseau	Naval Air Warfare Center
Rich Widmann	Hughes Aircraft Company, Electro-Optical Systems

INCOSE SECAM, Version 1.40, dated May 1995:

Blake Andrews	Rockwell Collins Air Transport Division
Tom Bachand	Mitre
Charles Baird	Systems Integration Software, Inc.
John Blyler	Tri-Cities Local INCOSE Chapter
LeRoy Botten	Computer Sciences Corporation
Lillian Brantley	Hughes Aircraft Company, Electro-Optical Systems
Gary Combs	Hughes Aircraft Company, Electro-Optical Systems
Richard Connett	Hughes Aircraft Company, Electro-Optical Systems
Dick Cramond	TRW
Ken Crowder	Boeing Information and Support Services
M. M. (Mac) Dillsi	MACTEC/Vectra GSI
Richard Garrison	Westinghouse Hanford
Gary Garside	Hughes Aircraft Company, Electro-Optical Systems
Paul Gartz	Boeing Commercial Airplane Company
Jerry Fisher	Loral Federal Systems
Norm Geis	Hughes Aircraft Company, Defense Systems
Charles Griner	Westinghouse Hanford
Mike Gross	Hughes Aircraft Company, Defense Systems
Dan Francis	Department of Energy, Hanford Facility
George Friedman	Past President, INCOSE
Mike Liewald	Hughes Aircraft Company, Electro-Optical Systems
Bob Lightsey	Defense Systems Management College
Doug Low	Hughes Aircraft Company, Electro-Optical Systems
Bill Mackey	Computer Sciences Corporation
Thomas McCann	Hughes Aircraft Company, Defense Systems

Brian McCay	Mitre
John Nyland	United Defense L.P.
Bob Olson	Naval Air Warfare Center - Weapons
Dave Olson	Hughes Aircraft Company, Electro-Optical Systems
Ron Olson	GTE Government Systems
Larry Pohlman	Boeing Information Systems
Frank Pohlner	Hughes Aircraft Company, Electro-Optical Systems
Dave Preklas	GTE Government Systems
Donna Rhodes	Loral Federal Systems
Terry Ross	Battelle Pacific Northwest Laboratory
Cho Shimizu	Boeing Commercial Airplane
Joseph Simpson	Tri-Cities Local INCOSE Chapter
Mary Simpson	Battelle Pacific Northwest Laboratory
George Stern	TASC
Sue Stetak	Hewlett Packard
Art Stone	Rockwell
Steve Tavan	Mitre
Michael Townsend	Naval Air Warfare Center - Indianapolis
Rich Widmann	Hughes Aircraft Company, Electro-Optical Systems
Ann Wilbur	Loral Western Development Laboratories

INCOSE SECAM, Version 1.50, dated April 1996:

Sam Alessi	US Department of Agriculture, North Central Soil Conservation Research Laboratory
Blake Andrews	Rockwell Collins Air Transport Division
Eugene Antonier	Hughes Aircraft Company, Electro-Optical Systems
Bruce Ammerman	Boeing Defense and Space Group
Eileen Arnold	Rockwell Collins General Aviation Division
Don Barber	Honeywell Air Transport System Division
Wayne Brandt	Rockwell Collins Air Transport Division
David Brenchley	Battelle, Pacific Northwest National Laboratory
Ross Cairns	Rockwell Collins General Aviation Division
Rhonda Coen	Hughes Aircraft Company, Electro-Optical Systems
Ken Crowder	Boeing Defense and Space Group
Steve Cuspard	Boeing Defense and Space Group
Jim Eklund	US Department of Agriculture, North Central Soil Conservation Research Laboratory
Dan Francis	Department of Energy, Richland Operations
George Friedman	Past President, INCOSE
George Hudak	AT&T Corporation
Curt Johansen	Physio-Control Corporation
Doug Low	Hughes Aircraft Company, Electro-Optical Systems
William Mackey	Computer Sciences Corporation
Brian Mar	University of Washington
John May	Boeing Defense and Space Group
Bill Miller	AT&T Corporation
Gar Norman	Westinghouse Hanford Company
John Nyland	United Defense L.P., Armament Systems Division
Dave Olson	Hughes Aircraft Company, Electro-Optical Systems
Frank Orsag	Westinghouse Hanford Company
Roger Orth	Boeing Defense and Space Group
Richard Pariseau	Naval Air Warfare Center
Rich Pavlik	Honeywell Commercial Avionics Systems Division
Ian Philpott	Rockwell Collins Air Transport Division
Larry Pohlman	Boeing Information Systems

Frank Pohlner	Hughes Aircraft Company, Electro-Optical Systems
Jock Rader	Hughes Aircraft Company, Radar and Communication Systems
R. Allen Ray	United Defense L.P., Armament Systems Division
George Richman	Hughes Aircraft Company, Electro-Optical Systems
Jack Ring	Innovation Management
Barney Roberts	Futron Corporation
Nancy Rundlet	QSS Inc.
Vince Saladin	Department of Energy, Richland Operations
Mark Sampson	TD Technologies, Inc.
Salley Schneider	US Department of Agriculture, IAREC, Systems Research Group
Michelle Sibernagel	Battelle Seattle Research Center
Joseph Simpson	Tri-Cities Local INCOSE Chapter
Mary Simpson	Battelle, Pacific Northwest National Laboratory
Doug Stemm	Hughes Aircraft Company, Electro-Optical Systems
Richard Stevens	QSS Ltd (UK)
Steve Tavan	Mitre
Jesse Teal	TRW Systems Integration
Michael Townsend	Naval Air Warfare Center, Indianapolis
Leng Vang	US Department of Agriculture, North Central Soil Conservation Research Laboratory
Ward Voorhees	US Department of Agriculture, North Central Soil Conservation Research Laboratory
Guy Wagner	Honeywell Defense Avionics Systems Division
Rich Widmann	Hughes Aircraft Company, Electro-Optical Systems
John Worl	Battelle Seattle Research Center

The current management of the INCOSE is:

INCOSE Board of Directors

Executive Officers:

Eric Honour	Harris Corp	President
Bill Schoening	McDonnell Douglas	President-Elect
V.A. (Ginny) Lentz	United Technologies Research Center	Past President
William D. Miller	AT&T Corporation	Secretary
Michael S. Wood	The Boeing Co	Treasure

Regional Directors:

Elliot Axelband	USC (II)
Peter Brook	Defense Research Agency (VI)
Don Clausing	MIT (IV)
Harry Crisp	Naval Surface Warfare Center (V)
Ken Crowder	The Boeing Co (I)
Robert Halligan	Technology Australasia Pty Ltd (VI)
Tom Kabaservice	Harris Corp (V)
Dorothy A. Kuhn	MCI (III)
Nancy Leveson	University of Washington (I)
Ron Olson	ZETA (II)
Jim Ray	ATCOM (III)
Nancy Rundlet	QSS Corp (IV)

Directors At Large:

Brian McCay	QSS Corp
Ken Ptack	PRC Inc
Mary Simpson	The Boeing Co

INCOSE Technical Board

The Technical Board sets the technical goals for INCOSE and is the primary approving authority for INCOSE technical products. This Board oversees the six technical committees and their special interest groups.

Donna Rhodes	Lockheed Martin Corporation	Chairman
John Velman	Hughes Aircraft Company	Education and Training
Bill Mackey	Computer Sciences Corp	SE Applications
Rich Widmann	Hughes Aircraft Company	Measurement
Mark Sampson	TD Technologies	Modeling and Tools
Rick Harwell	Lockheed Martin Corporation	SE Management
John Snoderly	Defense Systems Mgmt College	Processes and Methods

Board of Directors Representatives:

Harry Crisp	Naval Surface Warfare Center
Robert Halligan	Technology Australasia Pty Ltd

Liaisons:

Terry Robar	Otis Elevator	TB Assistant
Robert Halligan	Technology Australasia Pty Ltd	Australia Liaison
TBA		UK Liaison
Dona Lee	Strategic Insight	Communications Committee Liaison

INCOSE Corporate Sponsors and Advisory Board Members

The CAB is composed of senior representatives from companies that have provided financial support to INCOSE.

Mary J. Simpson	The Boeing Co	Chairman
Ken Ptack	PRC	Co-Chairman
Dwight Abbott	The Aerospace Corp	
Charles C. Schmidt	Allied Signal, Inc	
Carol Gutierrez	Ascent Logic Corp	
Mark B. Triplett	Battelle-PNNL	
Earl F. Weener	The Boeing Co	
David V. Burke	C.S. Draper Lab, Inc	
Myron J. Ross	GTE	
Don Barber	Honeywell Corp	
Billy H. Saeki	Hughes Aircraft Company	
Thomas J. Duffy	Lockheed Martin Corporation	
Richard J. Schwadron	McDonnell Douglas	
Lynda Rosa	The MITRE Corp	
Jim Cloud	Motorola	
Edward Conroy	Northrop Grumman Corp	
Paul M. Clemente	Raytheon Corp	
Robert J. Tibor	Rockwell Collins Avionics	

John D. Grimm	Texas Instruments	
Frank Ridolphi	TRW	

EPIC SE-CMM, Version 1.0, dated December 1994:

The authors of the EPIC SE-CMM v1.0 were as follows:

Roger Bate	Software Engineering Institute	Chief Architect
Suzanne Garcia	Software Engineering Institute	Project Leader
James Armitage		
Kerinia Cusick	Hughes Space & Communications Company	
Robert Jones	Loral Federal Systems	
Dorothy Kuhn	Texas Instruments	
Ilene Minnich	Hughes Aircraft Company	
Hal Pierson	Software Productivity Consortium	
Tim Powell		
Al Reichner	Loral Space and Range Systems	
Curtis Wells	Lockheed Martin Corporation	

EPIC SE-CMM, Version 1.1, dated November 1995:

The authors of the EPIC SE-CMM v1.1 were as follows:

Roger Bate	Software Engineering Institute	Chief Architect
Dorothy Kuhn	Texas Instruments	Product Manager
Curt Wells	Lockheed Martin Corporation	Product Manager
James Armitage		
Gloria Clark	Hughes Space & Communications Company	
Kerinia Cusick	Hughes Space & Communications Company	
Suzanne Garcia	Software Engineering Institute	
Mark Hanna }		
Robert Jones	Loral Federal Systems	
Peter Malpass	Software Engineering Institute	
Ilene Minnich	Hughes Aircraft Company	
Hal Pierson	Software Productivity Consortium	
Tim Powell		
Al Reichner	Loral Space and Range Systems	

The current management of EPIC is the EPIC Steering Group:

Lisa Finneran	Software Productivity Consortium	
William Gibbs	General Dynamics Corporation Martin	
Martin Leyba	Raytheon Systems Company	
Ilene Minnich	SECAT LLC	
Arthur Pyster	Federal Aviation Administration	
Scott Reed	Software Engineering Institute	
Mark Schaeffer	U.S. Department of Defense	
Joan Weszka	Lockheed Martin Corporation	Chair
Merle Whatley	Raytheon Systems Company	
Thomas Rhodes	NIST	
Colin Tully	European Software Process Improvement Foundatio	

Index

—D—

—E—

—F—

—G—

—I—

—K—

—L—

—M—

—N—

—O—

—P—

—Q—

—R—

—V—

—W—

Feedback for Improvements to Model

Suggestions for improvements to this standard should be directed to:

Engineering Department
2500 Wilson Boulevard
Arlington, VA 22201