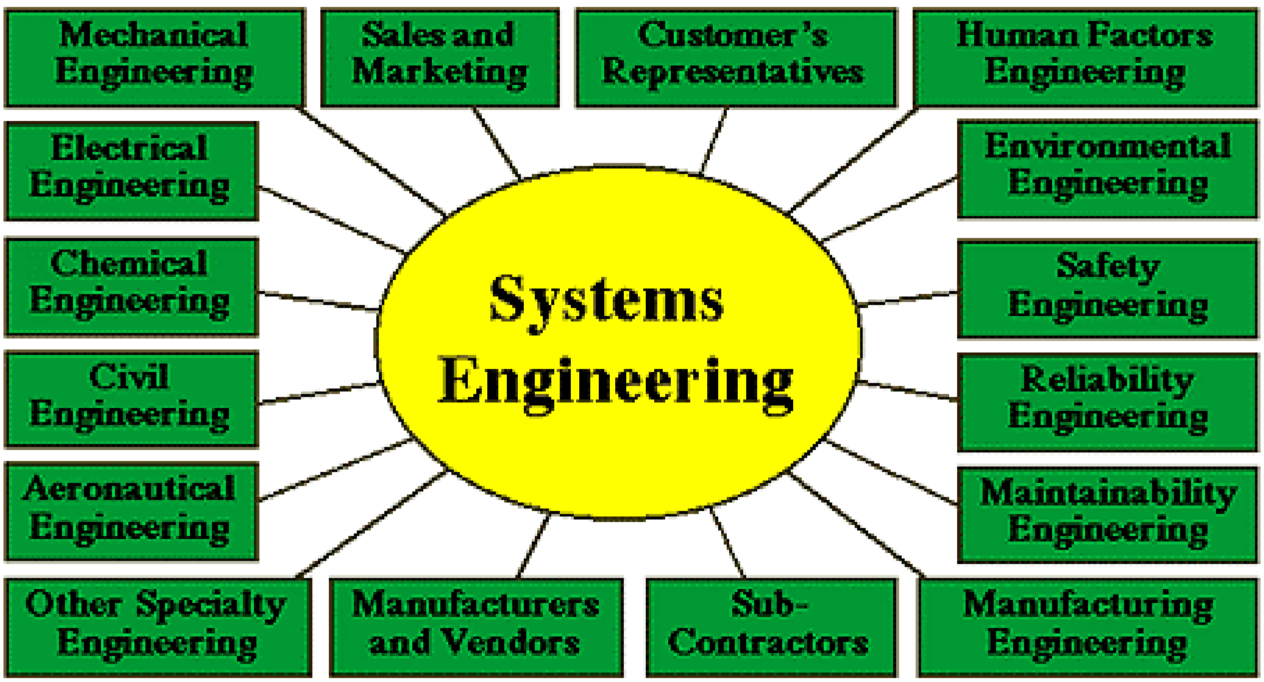
Systems Engineering Guidebook

A Guide for Developing, Implementing, Using and Improving Appropriate, Effective and Efficient Systems Engineering Capabilities



First Edition – April 2018

Published by Hall Associates LLC, Toney, Alabama

Library of Congress Cataloging-In-Publication Data:
Hall, David C.,
Systems Engineering Guidebook: A Guide for Developing, Implementing, Using and Improving Appropriate, Effective and Efficient Systems Engineering Capabilities
Includes bibliographical references and appropriate metrics
ISBN – 978-0-692-09180-7
Library of Congress Control Number: 2018938448

Table of Contents

Preface

Why this document? Note that most Systems Engineering (SE) definitions describe SE as: ***Systems Engineering*** is an **interdisciplinary approach and means** to enable the realization of successful *systems*. ... ***Systems Engineering*** **integrates all the disciplines and specialty groups** into a team effort forming a structured development process that proceeds from concept to production to operation[1]. NASA systems engineering is defined as a "methodical**, multi-disciplinary** approach for the design, realization, technical management, operations, and retirement of a system". A "system" is the combination of elements that function together to produce the capability required to meet a need. However, all current and expected work by Systems Engineering professionals seems to address only the individual processes and activities described as part of SE. The INCOSE Handbook, version 4.0 does outline some of the integration requirements for the Systems Engineering discipline. My research on this has shown that when it comes to trying to develop and implement *appropriate* overall SE for an organization or company, it doesn't happen. All processes and activities are implemented and used in a siloed fashion, with very little attempt to ensure that everything is integrated or even measured. This document defines and shows how to actually use the *SE Process* to more effectively and efficiently develop, implement, use *and improve* overall SE capabilities. In other words, establish a Systems Engineering Implementation and Use ***Project*** that accomplishes those actions. Use of the SE Process on this Project then allows optimization of SE capability, increases the overall Return on Investment of using SE and enables management to determine how effectively their personnel are complying with SE policy/directives. To better understand the makeup of this Guide, recommend study on Systems Thinking. There are some useful references in the Reference section.

Projects like this demand a lot of time and extensive experience since they go outside of the "normal" Systems Engineering set of processes into the enabling activities. My sincere gratitude and appreciation to colleagues who contributed to this book. My special thanks to Rebecca Falcon and Charlie Spillar who worked with me throughout the entire project and really aided in its completion.

Comments

Comments for update/revision are welcome from any interested party. Suggestions for change in the document should be in the form of a proposed change of text, together with appropriate supporting rationale. Please use the feedback form that is provided at the end of this document. Comments and requests for interpretations should be addressed to the author:

David C. Hall
Toney, AL 35773
Dave.hall@hallassociateshsval.com

[1] *www.incose.org/AboutSE/WhatIsSE*C

Systems Engineering Guidebook
A Guide for Developing, Implementing, Using and Improving Appropriate, Effective and Efficient Systems Engineering Capabilities

1.0 Introduction

This Systems Engineering Guidebook is a template describing "What" to do to successfully employ the Systems Engineering Process to implement and apply appropriate (effective and efficient) Systems Engineering processes and activities. Using this guide to implement Systems Engineering capability also allows use of the Systems Engineering Maturity Model described in Document HA2017-02. The Appendices (Section II) contain a description of the members of a SEIPT, a suggested metrics spreadsheet and list of reference documents that anyone can get and review if necessary. Note that once you develop an integrated timeline, you can implement in stages – implementing one process or activity at a time based on that integrated implementation time line. Implementation can be accomplished this way but it may be more efficient (depending on available resources and necessary culture changes) to implement multiple processes and activities at the same time. Standards and handbooks address life cycle models and SE processes and activities[2] that may or may not fully apply to a given organization and/or project. The objective of this Guide is to ensure that the Systems Engineering process and activity set meet the needs of the Enterprise/Organization/Program/Project while being scaled to the level of rigor that allows the system life cycle activities to be performed with an acceptable level of risk and to be measured. All SE processes and activities must be tailored to a rigorous application that provides an appropriate level based on need. While all SE processes and activities apply to all life cycle stages, tailoring determines the process/activity level that applies to each stage, **and that level is never zero**. There is always some effort in each process and activity in each stage. At the enterprise or organizational level, the tailoring process adapts external or internal standards in the context of the enterprise or organizational processes to meet the needs of the

---

[2] Note that the most useful overall are the INCOSE Systems Engineering Handbook and Systems Engineering Body of Knowledge. See reference list for additional documents.

enterprise/organization. At the program level, the tailoring process should adapt enterprise or organizational SE processes and activities to the unique needs of the program[3].

The objective of this Guide is to use the Systems Engineering Process on your System Engineering Implementation, Use and Update Project to implement, use, measure and improve your Systems Engineering capabilities. It also forces you to pull together the interdisciplinary processes and activities used in most enterprises and organizations. It covers the fundamental elements and lessons learned in systems engineering and used in developing, implementing, using, measuring and analyzing an overall Systems Engineering process/activity set **tailored** to the needs of an enterprise, an organization or a specific program/project. The Guide requires proper integration of disciplines and specialties – whichever ones are necessary and appropriate for a particular product (**in the broadest definition of "product**"). It is recommended that such implementation be done at the enterprise/organizational level and modified as necessary for specific programs rather than being done separately for each program. Many organizations currently document a program's systems engineering process in the "Systems Engineering Management Plan." Details for a SEMP (SEP) are described in the INCOSE Handbook, SEBok, IEEE 1220, DOD (EIA 632) Systems Engineering Standards and many other documents (Section III - Reference List). However, historical data and lessons learned have made it apparent that it is necessary to develop a **Systems Engineering Process Implementation Plan** (following the guidelines in this Guide) to cover all the necessary steps in accomplishing an appropriate tailoring of the SE processes and activities, implementing these tailored processes/activities, executing them, measuring the outcomes and providing continuous improvement. It has also been found that if an enterprise or organization develops (and uses) a Systems Engineering Integrated Process Team (SEIPT) to work an SE implementation, use and update project, the likelihood of success increases significantly (See Appendix A for the required expertise of personnel on a SEIPT).

The purpose of Systems Engineering is to increase a program's/project's likelihood of success and reduce the risk of failure[4]. All programs require either formal or informal systems

[3] This assumes the enterprise has a complete set of SE processes and activities defined and a program simply needs to tailor to fit their specific needs. If it is discovered that the enterprise does not have a required processes or activity, then the program should use this methodology to incorporate the new process or activity in the enterprise level.

engineering. Systems Engineering is not the sole responsibility of systems engineers: all engineers and developers must practice systems engineering. This Guide is intended to be used as a road map for integrating modern systems engineering disciplines and modern product realization processes and activities into programs. The Project implementation process outlined in this Guide is based on the overall INCOSE Systems Engineering process. Therefore, the steps described are the steps required for a typical development program/project. These steps are equally valid for development of new products, modifications of existing products, and replacement of components or subsystems within existing products. Additional systems development activities such as project and program management are also included. Note that Systems Engineering and Project Management overlap considerably (see figure 2). A good systems engineering process or activity contains both technical and management functions. It is the responsibility of the Systems Engineer and the Program Manager to coordinate these activities and eliminate duplications.

It is extremely important that you understand where your product is in its life cycle. You can develop and implement systems engineering for your specific life cycle phase but it is more efficient to tailor and implement systems engineering capable of working throughout the entire product life cycle. The system life cycle has seven general phases: (1) discovering system requirements, (2) creating and evaluating concepts, (3) design and development, (4) system verification, (5) system production, (6) operation, maintenance and modification, and (7) retirement, disposal, recycle, and replacement. The exact definitions and descriptions of the system life cycle can be different for different industries, products and customers but all variations address the above seven phases.

This Guide describes the activities recommended for the DISMI System Engineering Process Implementation Project (see figure 1). One might conclude from this Guide, though incorrectly, that the steps must occur in a linear sequence. In general, it is true that step n will usually begin before step n+1 begins. (In the terminology of program management, this is a start-to-start constraint.) With real-world programs, many of the steps will proceed in parallel. Some groups

---

[4] In addition to the INCOSE definition, one of the more comprehensive SE definitions comes from the US Air Force: Systems Engineering is the discipline encompassing the entire set of scientific, technical and managerial processes needed to conceive, evolve, verify, deploy and support an integrated Systems of Systems capability to meet user needs across the life cycle.

of steps will actually form iterative loops. Occasionally other constraints may change the actual sequence of the steps. However, the steps in the order described here are a good guide. The definitions of the stages in this document are consistent with the definitions of the phases or stages used in industry, academia, Department of Defense, and Department of Energy literature. The areas of concern (risks to properly developing, implementing, using, measuring and analyzing Systems Engineering processes and activities) within your enterprise or organization (or for individual programs/projects) are:

1. **People:** Who are your systems engineers? Is systems engineering a job title, or does it describe anyone who wants to think about the larger system that a product fits into, or only people with "Systems Engineering" degrees, or something certifiable by INCOSE? Note that excellence comes from people, not processes.

2. **Culture:** What is your current management and work culture and how resistant is it to change? How much change is going to be required? References on developing a Culture Change Management Plan are provided.

3. **Value:** What is the value to your organization or company of performing optimized and effective systems engineering? What are the benefits of systems engineering you are expecting (what are your goals/objectives?).

4. **Training:** How should your system engineers and other personnel be educated? What classroom and on-the-job training is important?

5. **Tools:** What tools do your systems engineers use? What tools can provide necessary support for everything systems engineering does in an integrated manner?

6. **Measurement and Assessment:** How do you measure systems engineering processes/activities? How do you assess a research and development organization, a maintenance organization, or an order fulfillment organization against a systems engineering model?

7. **Standards:** Who should use systems engineering standards (or domain best practices), and how should they use them? Do the various standards apply differently to different implementations of systems engineering? How do systems engineering standards apply to a small company making piece parts, consumer goods or services?

8. **Future:** How is your systems engineering capability expected/required to change in the future?

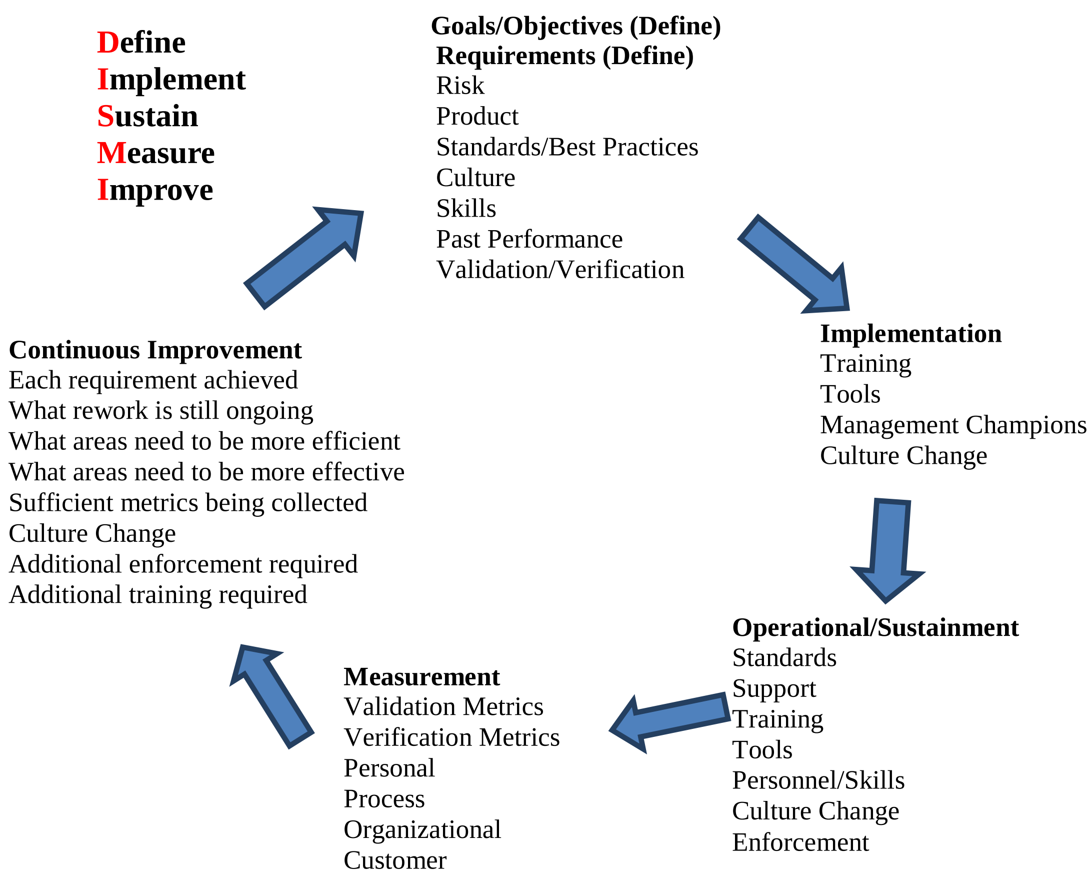


Figure 1: Systems Engineering Implementation Using the DISMI SE Process

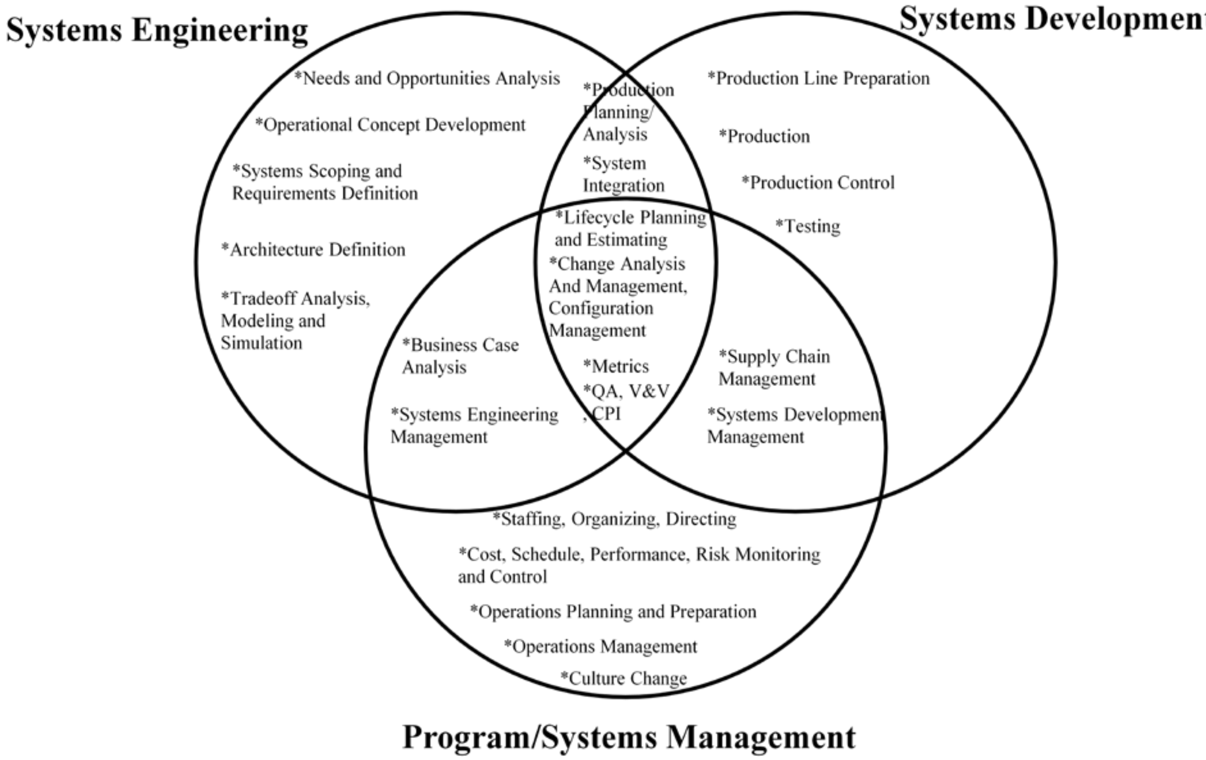


Figure 2: Systems Engineering Interfaces
(SEBoK, http://sebokwiki.org/wiki/Systems_Engineering_Overview)

2.0 Definition Phase

2.1 Define and Describe Goals and Objectives

Goals and objectives establish criteria and standards against which you can determine performance. You need to identify your goals and objectives for implementing and operating an effective and efficient Systems Engineering process. Note that a goal is a broad statement about the long-term expectation of what should happen as a result of implementing Systems Engineering (the desired result). These Goals serve as the foundation for developing your specific objectives. Objectives are statements describing the results to be achieved, and the manner in which they will be achieved. You usually need multiple objectives to address a single goal. Objectives should be

1. **Specific:** includes either “who”, “what”, or “where”. Use only one action verb to avoid issues with measuring success,
2. **Measurable:** focuses on “how much” change is expected,

3. **Achievable:** realistic given organizational or company resources and planned implementation,

4. **Relevant:** relates directly to goals,

5. **Time-bound:** focuses on "when" the objective(s) will be achieved.

6. **Doable:** Ensure that all Goals and Objectives take into account the interactions between Systems Engineering processes/activities and Program Engineering/Management processes and activities.

Examples of Systems Engineering Goals

1. Systems Engineering (SE) must establish the technical framework for delivering materiel or service capabilities to the customer and assure that the design addresses the actual problem.
2. SE must provide the foundation upon which everything else is built and support program success. The desired design is technologically possible.
3. SE must ensure the effective development and delivery of capability through the implementation of a balanced approach with respect to cost, schedule, performance, and risk using integrated, disciplined, and consistent SE activities and processes regardless of when a program enters its life cycle.
4. SE must enable the development of engineered resilient systems that are trusted, assured, and easily modified (agile).
5. The SE process must be comprehensive and reduce the likelihood of large-scale redesign.
6. The SE process must fit with existing enterprise or organizational processes and procedures (for ease of implementation, this should also include customer processes and procedures) or recommend necessary changes to those processes and procedures.
7. The SE capabilities required must be available within 6 months.

Examples of Systems Engineering Objectives That Relate to the Goals

1. Systems Engineering must support development of realistic and achievable program performance, schedule, and cost goals[5].
2. Systems Engineering must provide the end-to-end, integrated perspective of the technical activities and processes across the product life cycle, including how the product (system) fits into a larger system of systems (SoS) construct.
3. Systems Engineering must emphasize the use of integrated, consistent, measureable and repeatable processes to reduce risk while maturing and managing the product baseline. The final product baseline forms the basis for production, sustainment, future changes, and upgrades.
4. Systems Engineering must provide insight into product life-cycle resource requirements and impacts on human health and the environment.
5. SE must identify the products of all processes and activities.
6. SE must identify all process dependencies – goes into and comes out of. (See reference to N2 diagram)
7. SE Implementation Plan must define and provide guidance on executing each process and activity.

2.2 Define and Develop Process/Activity Requirements That Address the Objectives

(**One Requirement for Each Appropriate**[6] individual SE process and activity based on the following) Once you have developed all necessary goals and related objectives, you then need to develop the requirements for each appropriate process and activity that relate to those goals and objectives (see figure 3). Implementation requirements analysis is critical to the success or failure of a systems engineering implementation and operational program. As with any set of requirements, these implementation requirements should be **documented, actionable, measurable, testable, traceable, related to identified objectives and defined to a level of detail sufficient for process implementation and able to be validated and verified.**

[5] Note that your objectives, if written succinctly, can be incorporated into your Program Schedule as milestones.
[6] Not necessarily **all** SE processes/activities but only those needed for this specific enterprise/organization/program. If you decide to leave out a process/activity, you should insert a rationale for each decision as you may need to add these processes/activities in the future.

Remember that if you cannot ***measure it***, you cannot **control it** and if you cannot **control it**, you cannot **manage it**. Each process and activity requirement should be based on the following (see Requirements Engineering and Management References in Section 2):

1. Standards/Contract/Best Practices/Other
2. Product(s) and life cycle stage
3. Validation/Verification requirements versus capabilities
4. Current enterprise/organizational or company culture and expected changes required
5. Required tailoring based on "product", contract, etc.

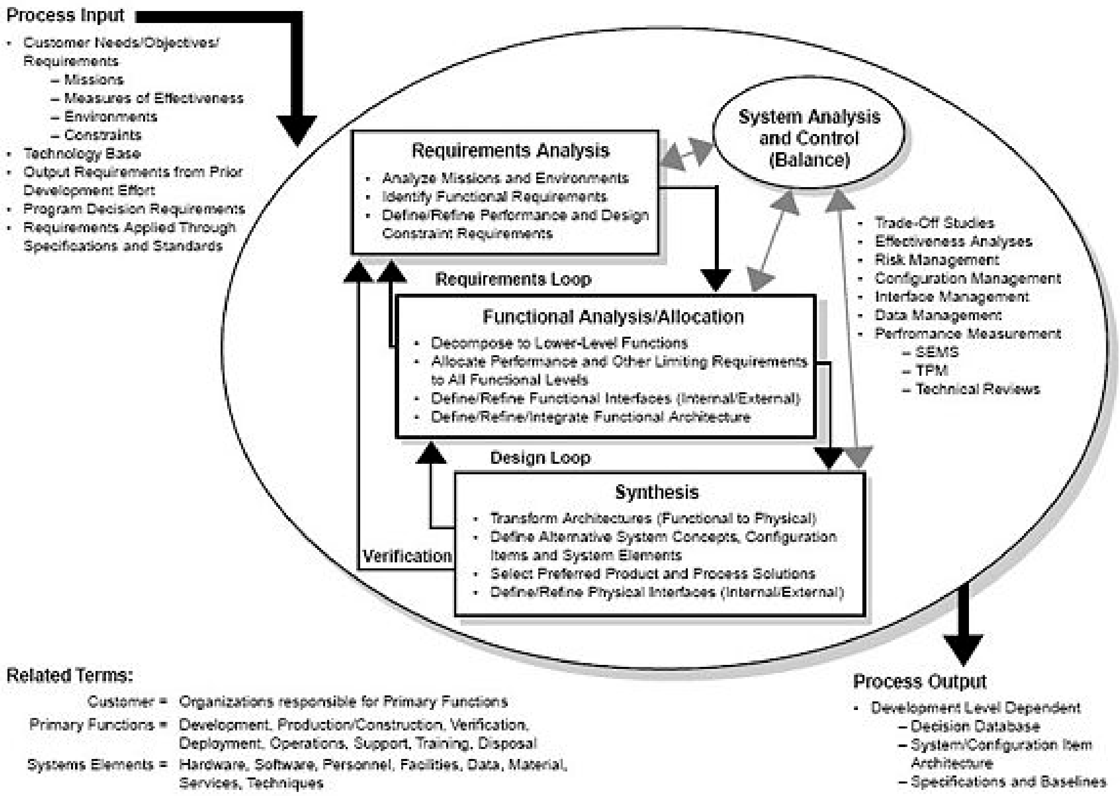


Figure 3: Develop the Appropriate Requirements (based on the ANSI/EIA 632 Egg Diagram)

Tailoring Activities

It is normally necessary to tailor each requirement to fit your specific domain and enterprise/organization/program portfolio. The following are areas you need to address to successfully tailor your requirements.

1. Identify and record the circumstances that influence tailoring.
2. Identify tailoring criteria for each life cycle stage - Establish the criteria to determine the process and activity level that applies to each life cycle stage.
3. Take due account of the life cycle structures recommended or mandated by standards, guides or best practices.
4. Obtain input from parties affected by the tailoring decisions.
5. Determine process or activity relevance to cost, schedule, and risks.
6. Determine process and activity relevance to system integrity.
7. Determine quality of documentation needed.
8. Determine the extent of review, coordination, and decision methods.
9. Make tailoring decisions.

2.2.1 Requirements Engineering and Management:

The purpose and scope of a Requirements Engineering and Management requirement is to define the process of developing, documenting, analyzing, tracing, prioritizing and agreeing on requirements and then controlling change (Requirements Management is managing changes to requirements) and communicating to relevant stakeholders. It is a continuous process throughout a program/project. The Requirements Management and Engineering process analyzes customer and stakeholder needs, generates/develops requirements, performs functional analyses, derives requirements, ensures requirements quality, allocates requirements, controls requirements, maintains requirements database, develops and implements Requirements Management Plans and develops measures of effectiveness and performance. A requirement is a capability to which a project outcome (product or service) should conform. The purpose of requirements management and engineering is to ensure that an organization documents, verifies, and meets the needs and expectations of its customers and internal or external stakeholders. Requirements management and engineering begins with the analysis and elicitation of the objectives and constraints of the organization. It must also include supporting planning for requirements, integrating requirements and the organization for working with them (attributes for requirements), as well as relationships with other information delivering against requirements, and changes for these. Based on history, this requirement should also include an explicit subprocess for the activities of Requirements Management and Requirements Engineering. These activities should include receiving the change requests from the stakeholders, recording the received change requests, analyzing and determining the desirability and process of implementation, implementation of the change request, and quality assurance for the implementation and closing the change request. Then the data of change requests be compiled, analyzed and appropriate metrics are derived and dovetailed into the organizational knowledge repository.[7],[8]

Example Requirement: *The standard to be used for this project is ANSI/IEEE Guide to Software Requirements STD 830-1984. All aspects of this standard shall be implemented unless specifically tailored out. If any are tailored out, rationale for the tailoring must be provided.*

Example Validation and Verification: *Verification and Validation of this requirement shall be by collection and analysis of the twenty two metrics defined as Requirements Metrics in Appendix C – Metrics Guide.*

2.2.2 Configuration Management

The purpose and scope of a Configuration Management Requirement is to define the configuration management policy, process, procedures and activities used to control and manage the development and modifications of products designed, developed, produced and maintained by Company/Organization. Configuration Management (CM) is the process of establishing and maintaining the technical integrity of a product throughout its life cycle by systematically identifying, controlling, and accounting for the product baseline and all changes made to the system. This Configuration Management Requirement may be tailored to fit product-unique configuration management requirements based on the life-cycle phase, complexity, size, intended use (including joint and combined interoperability), mission criticality, and logistic support of the product's Configuration Items (CIs). For a list of Configuration Management and related standards, see appendix D.

---

[7] Requirements Engineering normally has critical problems which can be due to lack of stakeholders' involvement in the requirements process. Lack of requirements management skills can also lead to bad requirements engineering. Unclear responsibilities and communication among stakeholders can also lead to bad requirements engineering.

[8] An Organizational Memory or Knowledge Repository is a computer system that continuously captures and analyzes the knowledge assets of an organization. It is a collaborative system where people can query and browse both structured and unstructured information in order to retrieve and preserve organizational knowledge assets and facilitate collaborative working.

Example Requirement: *The standard to be used for this project is ANSI/EIA-649-1998 National Consensus Standard for Configuration Management. All aspects of this standard shall be implemented unless specifically tailored out. If any are tailored out, rationale for the tailoring must be provided.*

Example Validation and Verification: *Verification and Validation of this requirement shall be by collection and analysis of the 12 metrics defined as Configuration Management Metrics in Appendix C – Metrics Guide.*

2.2.3 Risk Management

Risks affecting enterprises/organizations/programs can have consequences in terms of economic performance and professional reputation, as well as environmental, safety and societal outcomes. Therefore, managing risk effectively helps organizations to perform well in an environment full of uncertainty. Risk Management develops and implements Risk Management Plans, identifies risk issues, assesses risk issues, prioritizes risks, develops and implements risk mitigation and tracks risk reduction activities. Each risk management system must reflect the specific circumstances of an enterprise/organization as a generic approach usually not adequate. Nevertheless, risk management standards can provide useful support for designing and implementing a comprehensive and consistent risk management system.

Risk management is critical to program success for any program. The purpose of addressing risk on programs is to help ensure program cost, schedule, and performance objectives are achieved at every stage in the life cycle and to communicate to all stakeholders the process for uncovering, determining the scope of, and managing program uncertainties. Since risk can be associated with all aspects of a program, it is important to recognize that risk identification[9] is part of the job of everyone and not just the program manager or systems engineer. That includes the test manager, financial manager, contracting officer, logistician, and every other team member. If required, an organization can add an Opportunity Management process mirroring the Risk Management process.

[9] No current risk management standard or guide requires a risk baseline including all risk areas be established and managed. However, it is essential that such a risk baseline (covering all areas of program risk – technical, management, operational, external, enterprise and organizational) be established at the start of any risk management process.

Example Requirement: *The standard to be used for this project is ISO 31000:2009, Risk Management. All aspects of this standard shall be implemented unless specifically tailored out. If any are tailored out, rationale for the tailoring must be provided.*

Example Validation and Verification: *Verification and Validation of this requirement shall be by collection and analysis of the 9 metrics defined as Risk Management Metrics in Appendix C – Metrics Guide.*

2.2.4 Baseline Control

Program baselines should use one of the attached templates (see Appendix C – References) as guidance for developing and documenting any baseline activities. The definition of this work area is – all of the technical information needed to support a process/product throughout its life cycle. Baseline Control develops and implements Configuration Management Plans, establishes and updates baselines for requirements and evolving configurations/products, establishes and implements change control processes, maintains traceability of configurations, participates in Configuration Control Boards, participates in configuration item identification and status accounting, participate in functional and physical configuration audits. There are many different baselines required for a program but all of them are under configuration management. In configuration management, a "baseline" is an agreed description of the attributes of a product, at a point in time, which serves as a basis for defining change. A "change" is a movement from this baseline state to a ***next*** state. The identification of significant changes from the baseline state is the central purpose of baseline identification. A Baseline Change Control subprocess must also be developed and administered in accordance with one of the typical Baseline Change Control standards. Normally, baseline control is also directed by specific management policies/directives. Responsibilities and requirements for management, administration, and use of the technical, schedule, and cost baseline control system should be defined, including the process for preparing and implementing the baseline change request (BCR). There are numerous Change Control templates available on the Internet. Choose one or more as required and add to the configuration management requirement. This should come under the overall configuration management standard used.

2.2.5 Systems Engineering Technical Planning

Planning is one of the fundamental functions of systems engineering and management at any level. It provides the basis for the other systems engineering functions, particularly tracking and controlling. This systems engineering work area is concerned with the planning of programs. By program, we mean an undertaking typically requiring concerted effort that is focused on developing, manufacturing, operating or maintaining a specific product or products. SE Technical Planning is identifying program objectives and technical development strategy; preparing Systems Engineering Management Plans, Product Breakdown Structures, program Work Breakdown Structures, Integrated Master Plans, and Integrated Master Schedules; identifying program metrics including product technical performance measures and key performance parameters, identify program resource needs in terms of equipment, facilities, and personnel capabilities. It is useful to distinguish between the process by which plans are created (the planning process) and the product of that process (the plans). Most planning processes are very similar regardless of the organizational level at which the plan is applied. They usually differ in the personnel involved and the scope of the planned effort. Generally, all planning processes should include:

1. establishing the plan and its contents
2. establishing estimates of the resources required to carry out the plan
3. having those who will be bound by the plan review it for feasibility
4. establishing commitments to the plan

The planning process needs to be iterative and ongoing–after all, plans change. You should provide methodologies to update and revise as needed during a plan's lifespan.

Different types of plans address different purposes. Examples of program-oriented technical plans include program plans, software development plans, quality assurance plans, configuration management plans, test plans, communications plans and risk management plans. Although the contents of each plan should be tailored to fit its particular use, plans typically contain the following:

1. Goals: A goal is a statement of a desired state that will be achieved by the successful execution of the plan.
2. Strategies: A strategy is a description of a way to achieve plan goals.

3. Objectives: An objective describes a significant, measurable, time-related intermediate state that will be achieved as the plan is executed.
4. A set of activities to perform: An activity is an assignable, discrete step that helps achieve the specified objectives.
5. Resources allocated: The plan should include an assessment of the resources that the planned activities are allowed to consume (chief among which is time).

Other potential plan contents include responsibilities and commitments, work breakdown structures, resource and schedule estimates, risks, progress measures, relationships, and traceability to other plans. The most usable plans have a particular focus. Planning a complex task often requires a set of interrelated plans that might have these relationships:

1. Temporal relationships: Some plans might cover a time period that precedes or follows that of other plans.
2. Hierarchical relationships: Some plans contain subordinate details.
3. Relationships involving critical dependencies: Some plans depend on the execution of other plans.
4. Relationships based on a supporting infrastructure: Some plans depend on the existence of an organizational function–for example, a quality assurance or process group.

There are no specific standards for the SE Technical Planning work area but there are numerous guidelines and templates for various plans. For example, look at the templates in the Defense Acquisition Guidebook, Chapter 4: Systems Engineering, Section 4.3.2. Technical Planning Process or in NASA NPR 7123.1B, Appendix C. Practices for Common Technical Processes (see figure 4) for defining the scope of the technical effort required to develop, field, and sustain a system, as well as providing critical quantitative inputs to program planning and life-cycle cost estimates. Develop a Plan Template(s) that applies to all types of Plans your program must develop.

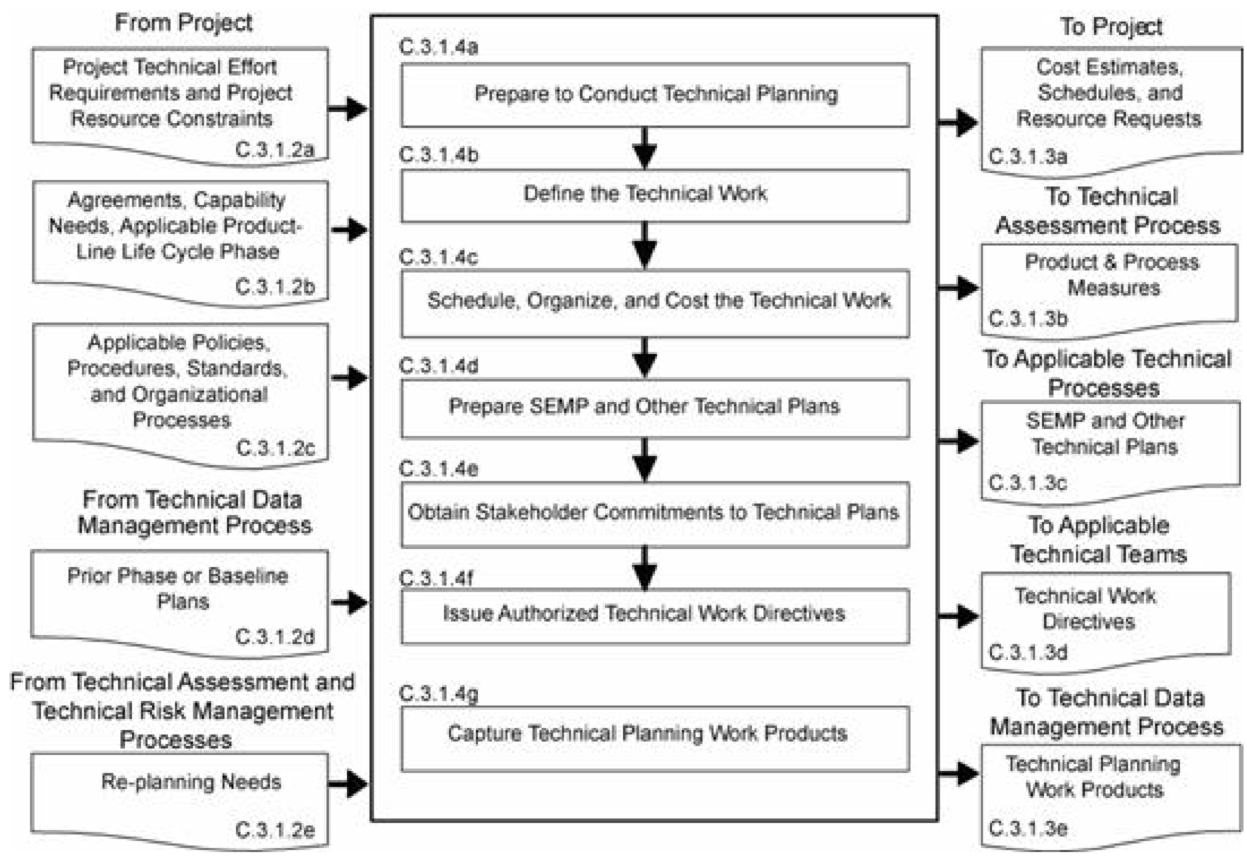


Figure 4: SE Technical Planning Activities

SE planning, as documented in a Systems Engineering Management Plan (SEMP), must identify the most effective and efficient process to deliver a capability, from identifying user needs and concepts through delivery and sustainment. SE event-driven technical reviews and audits must assess program maturity and determine the status of the technical risks associated with cost, schedule, and performance goals.

Example Requirement: *The SE Technical Planning Best Practices to be used for this program are shown in NASA NPR 7123.1B, Appendix C. Practices for Common Technical Processes. All aspects of this Best Practice shall be implemented unless specifically tailored out. If any are tailored out, rationale for the tailoring must be provided.*

Example Validation and Verification: *Verification and Validation of this requirement shall be by collection and analysis of the 4 metrics defined as Technical Planning Metrics in Appendix B– Metrics Guide. If these Plans are covered by Configuration Control, then various Configuration Management Metrics can be substituted for the Technical Planning metrics.*

2.2.6 Technical Effort Assessment

The Systems Engineer assists the Program Manager in planning and conducting the Technical Assessment process. This includes advising on technical reviews and audits, defining the technical documentation and artifacts that serve as review criteria for each review/audit, and identifying TPMs. Specific activities include:

1. Establishing event-driven technical planning
2. Identifying appropriate measures and metrics
3. Identifying performance measures to assess program health and technical progress
4. Conducting analyses to determine risk and to develop risk mitigation strategies
5. Conducting assessments of technical maturity, process health and stability, and risk to communicate progress to stakeholders and authorities at key decision points
6. Proposing changes in the technical approach to address risk mitigation activities
7. Advising the Program Manager regarding the technical readiness of the program to proceed to the next phase of effort
8. Obtaining independent subject matter experts as appropriate for reviews and audits

Technical Effort Assessment (measurement) is the method of collecting and providing information to Program Managers and Systems Engineers at predefined intervals for decision making. Technical Assessment Effort Metrics constitute the data that identify the need for improvement (i.e., the facts and trends of process performance) and provide a basis for assessing the improvements. The Technical Assessment process allows the Systems Engineer to compare achieved results against defined criteria to provide a fact-based understanding of the current level of product knowledge, technical maturity, program status, and technical risk. This assessment results in a better understanding of the health and maturity of the program, giving the Program Manager a sound technical basis upon which to make program decisions. Technical Effort Assessment collects, analyzes, tracks, and reports program metrics including product technical performance measures and key performance parameters; conduct audits and reviews; assess process and tool usage compliance; conduct capability assessments; recommend and implement process and product improvements. The Program Manager and Systems Engineer evaluate technical maturity in support of program decisions at the key event driven technical reviews and audits that occur throughout the acquisition life cycle. The Program Manager and Systems

Engineer use various measures and metrics, including Technical Performance Measures (TPM) and leading indicators, to gauge technical progress against planned goals, objectives, and requirements.

Example Requirement: *The Best Practices to be used for this project are shown in NASA NPR 7123.1B, Appendix C. Practices for Common Technical Processes. All aspects of this Best Practice shall be implemented unless specifically tailored out. If any are tailored out, rationale for the tailoring must be provided.*

Example Validation and Verification: *Verification and Validation of this requirement shall be by collection and analysis of the metrics defined as Technical Effort Assessment metrics in Appendix C – Metrics Guide.*

2.2.7 Architecture/Design Development

The trend today is to consider system architecture and system design as different and separate sets of activities, but concurrent and strongly intertwined. For the purposes of this Implementation Guide, we will consider these activities to be separate parts of the same process.

The purpose of system architecture activities is to define a comprehensive solution based on principles, concepts, and properties logically related and consistent with each other, identify baseline and alternate candidate concepts and architectures, prepare Trade Study Plans, conduct and document trade studies, evaluate and optimize candidate concepts and architectures, prepare system/solution description documents. The solution architecture has features, properties, and characteristics satisfying, as far as possible, the problem or opportunity expressed by a set of system requirements (traceable to mission/business and stakeholder requirements) and life cycle concepts (e.g., operational, support) and are implementable through technologies (e.g., mechanics, electronics, hydraulics, software, services, procedures, human activity). System Architecture is abstract, conceptualization-oriented, global, and focused to achieve the mission and life cycle concepts of the system. It also focuses on high level structure in systems and system elements. It addresses the architectural principles, concepts, properties, and characteristics of the system-of-interest. It may also be applied to more than one system, in some cases forming the common structure, pattern, and set of requirements for classes or families of similar or related systems.

System design is intended to be the link between the system architecture (at whatever point this milestone is defined in the specific application of the systems engineering process) and the implementation of technological system elements that compose the physical architecture model of the system. Design definition is driven by specified requirements, the system architecture, and more detailed analysis of performance and feasibility. It addresses the implementation technologies and their assimilation. Design provides the "how" or "implement to" level of the definition. Design concerns every system element composed of implementation technologies (for example mechanics, electronics, software, chemistry, human operations and services) for which specific engineering processes are needed. System design provides feedback to the parent system architecture to consolidate or confirm the allocation and partitioning of architectural characteristics and design properties to system elements. The purpose of the System Design is to supplement the system architecture providing information and data useful and necessary for implementation of the system elements. Design definition is the process of developing, expressing, documenting, and communicating the realization of the architecture of the system through a complete set of design characteristics described in a form suitable for implementation. System design includes activities to conceive a set of system elements that answers a specific, intended purpose, using principles and concepts; it includes assessments and decisions to select system elements that compose the system, fit the architecture of the system, and comply with traded-off system requirements. It is the complete set of detailed models, properties, and/or characteristics described into a form suitable for implementation.

Example Requirement Development *process* **–**

The following Best Practices have been chosen for this requirement:

1. *Choose and get approved an appropriate development lifecycle process to the project at hand. All other activities are to be derived from the chosen lifecycle process. For an example software development project a spiral-based methodology is chosen.*
2. *Gather and agree on requirements for the project.*
3. *Choose the appropriate architecture for your application. Apply well-known industry architecture best practices.*
4. *Keep the design as simple as possible.*
5. *Conduct periodic Peer reviews including all artifacts from the development process (including plans, requirements, architecture, design, code, and test cases).*

6. *Testing must be planned as an integral part of software development. Testing is to be planned and carried out proactively - test cases are to be planned before coding starts and test cases are to be developed while the application is being designed and coded.*
7. *Configuration management - knowing the state of all artifacts that make up your system or project, managing the state of those artifacts, and releasing distinct versions of a system – must be carried out throughout the process.*
8. *Establish Quality Priorities, a Defects Management activity and release criteria for the project.*
9. *A defect tracking system must be used that is linked to the source control management system.*

Example Validation and Verification: *Verification and Validation of this requirement shall be by collection and analysis of the chosen metrics defined as Software metrics in Appendix C – Metrics Guide.*

2.2.8 Qualification, Verification and Validation

Qualification is a process of assurance that the specific product, premises or equipment are able to achieve the predetermined acceptance criteria to confirm the attributes of what it is supposed to do. It is a process to demonstrate the ability to fulfill specified requirements. There are several types of qualification. You must first determine what qualification(s) is required for your product.

1. Installation Qualification (IQ) – Establishing confidence that process equipment and ancillary systems are compliant with appropriate codes and approved design intentions, and that manufacturer's recommendations are suitably considered. In other words: (1) installation of hardware and system software per the manufacturer’s instructions, or (2) in the cloud, the provisioning of a virtual machine per an approved procedure and the installation of system software per the manufacturer’s instructions

2. Operational Qualification (OQ) – Establishing confidence that process equipment and sub-systems are capable of consistently operating within established limits and tolerances. In other words: testing against the documented and approved requirements

and specifications (unit, string, and integration testing per the documented and approved system design specifications; and system testing per the documented and approved functional requirements).

3. Performance Qualification (PQ) – (1) process performance qualification: establishing confidence that the process is effective and reproducible, or (2) product performance qualification: establishing confidence through appropriate testing that the finished product produced by a specified process meets all release requirements for functionality and safety.

Validation

Validation is establishing documented evidence which provides a high degree of assurance that a specific process/activity will consistently produce a product meeting its predetermined specifications and quality attributes. It is establishing confidence that process equipment and sub-systems are capable of consistently operating within established limits and tolerances. Note that before accomplishing validation on a product, be sure that it has passed qualification[10].

Verification

Product (System) Verification is a set of actions used to check the correctness of any element, such as a product element, a product, a document, a service, a task, a requirement, etc. These types of actions are planned and carried out throughout the life cycle of the product. Verification is a generic term that needs to be instantiated within the context it occurs. As a process, verification is a transverse activity to every life cycle stage of the product. In particular, during the development cycle of the product, the verification process is performed in parallel with the product definition and product realization processes and applies to any activity and any product resulting from the activity. The activities of every life cycle process and those of the verification process can work together. The four fundamental methods of verification are Inspection,

[10] Adding to the confusion caused by these terms with similar and overlapping meanings, different organizations mix the terms and definitions. Some organizations refer to verification as validation. Some define verification as dynamic testing and validation as static testing (i.e., peer review). Others refer to testing as verification or qualification. And others refer to qualification as validation. What's important is not that we agree on terms, but that we understand all the activities associated with the validation of systems and ensure that they are performed.

Demonstration, Test, and Analysis. The four methods are somewhat hierarchical in nature, as each verifies requirements of a product or system with increasing rigor. Each enterprise and organization should clearly define each of the four primary verification methods: Test, Demonstration, Inspection, and Analysis.

Example Qualification Requirement[11]

The Standards to be used for this process are the ISO 9000 series standards:

1. *ISO 9001:2015 - sets out the requirements of a quality management system*
2. *ISO 9000:2015 - covers the basic concepts and language*
3. *ISO 9004:2009 - focuses on how to make a quality management system more efficient and effective*
4. *ISO 19011:2011 - sets out guidance on internal and external audits of quality management systems.*

None of these standards are tailored in any way.

Example Verification and Validation

Verification and Validation of this requirement shall be by collection and analysis of the metrics defined as Quality Metrics in Appendix C – Metrics Guide.

Example Verification Requirement

Develop and implement Verification Plans; develop verification requirements and pass/fail criteria; conduct and record results of qualification, verification, and validation efforts, and corrective actions; prepare requirements verification matrix and qualification certificates. Establish confirmation, through the provision of objective evidence, that specified requirements have been fulfilled. Provide specific identification of the element on which the verification action will be performed and identification of the reference to define the expected result of the verification action.

[11] If the enterprise or organization separates Qualification and Quality Assurance the following should be added as a Quality Assurance requirement: develop and implement a Quality Assurance Plan, perform quality audits, report quality audits, define and track quality corrective actions.

Example Validation and Verification:

Validation and verification shall be by collection and analysis of the metrics defined by Technical Performance Metrics and Validation/Verification metrics in Appendix C – Metrics Guide.

Example Validation Requirement

*Develop plans and metrics for evaluating the operational effectiveness, operational suitability, sustainability, and survivability of the system or system elements under operationally realistic conditions. Validation activities can be conducted in the intended operational environment(s) or on an **approved** simulated environment. Final validation shall consist of user operational testing on a production-representative product (system) in an operationally realistic environment.*[12]

Example Validation and Verification

Validation and verification shall be by collection and analysis of the metrics defined by Technical Performance Metrics and Validation/Verification Metrics in Appendix C – Metrics Guide.

2.2.9 Training

Systems engineering is NOT a rulebook. It is a set of principles (processes and activities) supported by methods deigned to deliver maximum benefits to stakeholders at minimum costs. A Systems Engineering training course set should be designed for personnel who currently perform, manage, control or specify the life cycle of products[13]. All courses and seminars can be delivered using a mixture of formal presentation, informal discussion, and extensive work shops which exercise key aspects of systems engineering on a single product or multiples of products through the life cycle. The desired result is a high degree of continuing learning on each Systems Engineering process and activity.

Competencies are the combination of knowledge, skills and abilities that contribute to individual and organizational performance. Any Systems Engineering developmental framework should be based on a rigorous set of competencies that personnel should have in order to perform their

[12] Tailoring Software Validation and Verification should be risk based on integrity levels.
[13] As noted before product is used in its broadest sense.

jobs. These competencies define the breadth and scope of the discipline and facilitate personnel development and assessment of individual knowledge and capabilities. Competencies developed form the foundation of any training program and should be under configuration control and reviewed and updated as appropriate.

A key step for managerial, engineering and technical personnel is to understand the requirements of their roles and the related competencies. Performance-level descriptions for each competency should be created to guide the overall development of individuals within the program and domain engineering disciplines.

Example Requirement

Conduct a Systems Engineering Training Needs and Skill Gap assessment based on the requirements set developed and an evaluation of existing Systems Engineering and other personnel skills/knowledge/experience. Develop and implement Systems Engineering Training Plans using the skills shown to be required for the team and each individual, develop and give training courses on enterprise/organizational Systems Engineering processes, activities and tools as required and assess results.

Example Validation and Verification

Validation and verification shall be by collection and analysis of the metrics defined by Systems Engineering Training metrics in Appendix B – Metrics Guide.

2.2.10 Systems Integration

In Systems Engineering, systems integration is defined as the process of bringing together the component subsystems into one product and ensuring that the subsystems function together as a product (system). In information technology, systems integration is the process of linking together different computing systems and software applications physically or functionally to act as a coordinated whole.

Example Systems Integration Requirement:

The following activities are to be accomplished in priority order:

1. *Define the technical integration strategy,*

2. *Develop required Integration Plans,*
3. *Develop integration test scripts,*
4. *Develop and implement integration test scenarios,*
5. *Conduct and document integration tests,*
6. *Track integration test results and retest status.*

Example Validation and Verification

Validation and verification shall be by collection and analysis of the metrics defined by Systems Integration metrics in Appendix B – Metrics Guide.

2.2.11 Specialty Engineering

In the domain of systems engineering, Specialty Engineering is defined as and includes the engineering disciplines that are not typical of the main engineering effort. More common engineering efforts in systems engineering such as hardware, software, and human factors engineering may be used as major elements in a majority of systems engineering efforts and therefore are not viewed as "special". Examples of specialty engineering include electromagnetic interference, safety, physical security, electromagnetic interference, electrical grounding, safety, cybersecurity, electrical power filtering/uninterruptible supply, manufacturability, and environmental engineering may be included defined as specialty engineering processes/activities where they have been identified to address special product implementations. However, if the specific product has a standard implementation of environmental or security for example, the situation is reversed and the human factors engineering or hardware/software engineering may be the "specialty engineering" domain.
The key take away is; the context of the system engineering project and unique needs of the project are fundamental when thinking of what are the specialty engineering efforts.

The benefit of citing "specialty engineering" in planning is the notice to all team levels that special management and science factors may need to be accounted for and may influence the project. Specialty engineering may also be cited by commercial entities and others to specify their unique abilities.

Example Specialty Engineering Requirement

Develop and implement Specialty Plans as part of, or an addendum to, the Systems Engineering Management Plan to cover reliability, maintainability, supportability, survivability, logistics support, security, safety, electromagnetic environmental effects, environmental engineering, packaging and handling, etc. Use specific standards from each of the areas to determine the process or activity.

Example Validation and Verification

Use metrics called out in the standard or best practices being used as a requirement for each area. It is expected that most of the metrics will already be covered by the list provided in the Metrics Guide, Appendix C.

2.2.12 Other Processes/Activities as needed/required

Describe other functions that you need to perform and can justify as system engineering activities. Provide a specific requirement and methods of validation and verification.

2.3 Development of Existing Enterprise/Organizational/ Program Processes and Activities View

Many modern organizations are functional and hierarchical; they suffer from isolated departments, poor coordination, and limited lateral communication. All too often, work is fragmented and compartmentalized, and managers find it difficult to get things done. In the broadest sense, Systems Engineering processes and activities can be defined as collections of tasks and activities that together - and only together – efficiently transform inputs into outputs. Within organizations, these inputs and outputs can be as varied as materials, information, and people.

Effective organization design considers five interrelated components

1. Clear vision and priorities and a cohesive leadership team
2. Clear roles and accountabilities for decisions and an organizational structure that supports objectives

3. Organizational and individual talent necessary for success and performance measures and incentives aligned to objectives
4. Superior execution of programmatic work processes and effective and efficient support processes and systems
5. High performance' values and behaviors and capacity to change

2.3.1 Current Organizational Culture

As you may know, enterprise/organizational/program culture varies based on numerous factors. Although most don't focus on the culture within the enterprise/organization/program, every one has a culture whether they like it or not. It is necessary to determine the existing culture of your enterprise/organization/program before you try to implement Systems Engineering capabilities since there will be numerous ways of personnel and policy pushback. Take a look at the following - each is part of the culture at your enterprise/organization/program:

1. Employees
2. Size
3. Past Performance (Lessons learned)
4. Environment (How do the company values play into the culture?)
5. Policies (Does the company address problems head on, how does it deal with new ideas?)
6. Procedures and activities
7. Mission
8. Values (Are workers encouraged to speak up and identify problems?)
9. Attitudes (How do employees within the organization handle conflict and change?)
10. Employee commitment
11. Communication
12. Common behaviors
13. Relationships (How well do employees work together?)
14. Leadership (Are employees rewarded for performance? How?)
15. Management (Do you have appropriate Management Champions?)

Once you understand your current culture and how the enterprise/organization/program reacts to change, then the necessary Culture Change Management to successfully implement SE capabilities involves the selection of strategies to facilitate the transition of individuals, teams, or entire enterprises/organizations/programs from a current state of operation to the new, desired state. More specifically, you must develop a process and set of techniques to manage the feelings, perceptions, and reactions of the ***people*** affected by the changes being introduced. This includes senior management as they hold the resources necessary for any change. The impetus of any change initiative is to improve some aspect of operations or longer term outcomes. Change projects result in new policies, processes, protocols, or systems to which staff must become accustomed and change management must be used to facilitate the transition. For successful culture change, attention must be given to both the “process” and “human” sides of change. The “process” side involves the specific project management related activities required for moving from the current to desired state (e.g., develop plans, build the infrastructure, change processes or systems, redefine job roles). The “human**”** side of change involves strategies to help employees impacted by the change understand and adopt it as a part of their jobs (e.g., alleviate staff resistance, meet training needs and secure buy-in).

Both aspects of change should be integrated and occur simultaneously for successful change, however, the change leader(s) may need to think of the “process” and “human” changes distinctly when assessing and addressing roadblocks. For example, an organization may have full employee buy-in for a particular change initiative but adequate resources and planning efforts have not been put in place to support the change. Alternatively, appropriate structures and processes may be in place but employees remain resistant to the initiative.

2.3.2 Current Organizational Policies, Procedures and Activities

The policies and practices within your organization have a significant impact on your culture and how difficult it will be to change that culture. They must be evaluated. Consider how:

1. Policies regarding pay scales, benefits, and opportunities to advance within the company all influence and help define your culture.
2. Rules related to discipline and dress code also influence the overall culture of the organization.

3. Practices and policies that govern how you do business, how you interact with suppliers, and how you serve customers – all help to shape your organizational culture.

The flow of information in your organization also strongly affects its culture. Consider the following questions regarding your information flows:

1. What kind of information is distributed in your company? Is it easy for your people to stay up-to-date about important information? This could include key metrics on company performance, both before and after a change.
2. How about information regarding your people? Do you highlight employee achievements, accolades and hobbies?
3. Where does the information flow in your organization? It can flow vertically from one level to another. It also flows horizontally, among co-workers. The informal organization created by information flow is as important as the formal organization flow.
4. What methods and mediums do you use to communicate with your people? Do you use email or an internal web portal to keep people up to date? How do you use face-to-face meetings to communicate with your people?

2.3.3 Change Management Methodologies and Processes

Once you understand the current culture in all of its aspects, there are numerous methodologies and processes for developing the Change Management part of the Implementation Plan. You need to choose the one that pertains to your specific enterprise/organization/program. (See Change Management references in appendix C). The areas necessary to be covered in the Change Management section are as follows:

1. Current Policy set
2. Current versus Required Skill Sets
3. Current Metrics Versus Required Metrics (Collection and automated tool sets)
4. Communication Methodology (Up and Down)
5. Develop Baseline Implementation and Operational Time Line (especially if going to implement one process at a time)

6. Baseline of Risks to Implementing the Plan (since almost all the risks/problems will be caused by people)

2.3.4 Tools Available for Culture Change

Companies and organizations have a wide range of tools at their disposal to align employee behavior with strategy and close the gap between their current and the target culture. To close this gap, your plan should make the most of seven critical levers that influence behavior and shape organizational culture. These levers represent a mix of hard and soft approaches that separately and in combination shape behavior. They enable organizations and companies to understand the forces shaping their current culture and to specify what needs to be changed in order to achieve and sustain the desired culture. The levers are as follows:

1. Leadership. Leaders' role-modeling behaviors; their manner of communication, especially in reinforcing desired behaviors; how they spend their time, manage their priorities, and interact with direct reports.
2. People and Development. The kind of personnel who are recruited and hired; opportunities for meaningful work and the kind of career paths and personal growth the organization/company enables; how talent is promoted and retained; the coaching that supervisors provide; the organization's/companies learning and development programs.
3. Performance Management. The key performance indicators that are used to define and track performance drivers, and policies and practices regarding compensation, benefits, reviews, promotions, rewards, and penalties, including the consequences of undesirable behavior.
4. Informal Interactions. Networks, the nature of peer-to-peer interactions, gatherings, and events, whether active communities of interest exist, whether people know whom to contact to access enterprise knowledge
5. Organization Design. Organizational structure, processes, and roles, decision rights, collaboration processes, units' relationship to headquarters, office layout and design
6. Resources and Tools. The projects that are funded, access to human resources, management systems, analytical tools
7. Values. The collective beliefs, ideals, and norms that guide peoples' conduct and help them adhere to priorities, especially when facing a difficult business problem.

2.3.4 Development of Required Training/New Capabilities

One basic lever to accomplish culture change is making sure that your personnel have appropriate knowledge and experience. Based on adult learning principles, the following are necessary steps for a successful personnel learning experience:

1. The goals of the Systems Engineering required training program are clear
2. All personnel **are involved** in determining the knowledge, skills and abilities to be learned
3. The work experiences and knowledge that employees bring to each learning situation are used as a resource
4. A practical and problem-centered approach based on real examples is used
5. New material is connected to past learning and work experiences
6. Personnel are given an opportunity to reinforce what they learn by practicing
7. The learning opportunity promotes positive self-esteem

A formal Training Plan should be developed based on having your personnel understand the overall concepts which are characteristic of a systems approach to engineering and individual Systems Engineering processes. Since personnel have various experiences and training, the Plan should cover all aspects of systems engineering processes regardless of current levels of experience. These aspects are:

1. understand the overall process elements, and their relationships, which collectively constitute the building blocks of systems engineering;
2. be able to perform the many of the more important techniques within system requirements analysis, development of physical solution, development of logical solution, evaluation of solution alternatives (trade-off studies) and design iteration;
3. be familiar with some of the principles and major techniques of engineering management in a systems project context;
4. have some basic capability to tailor the application of the systems engineering principles, processes and methods to different domains and application scenarios;
5. be capable of further learning in the field of systems engineering as necessary to achieve the goals and objectives as they evolve.

2.4 Required Development of Enterprise/Organization/Program

2.4.1 Desired Culture

The following are the minimum generic culture statements that should be examined when developing your desired Systems Engineering culture.

1. Managers should be seen as coaches and team leaders. Leadership is participative and flexible.
2. Organizational policies and procedures and training are developed to help people get the job done. All are periodically reviewed and changed as needed.
3. Information is readily shared. Conflicts are addressed openly and respectfully.
4. Productivity is measured by the results achieved against the approved goals and objectives.
5. There is a high level of trust that people will do the right thing and policies and procedures reflect this. Problems are dealt with as they occur.
6. Collaboration is freely entered into.
7. People get on-going feedback about their performance in a constructive, helpful manner.
8. People are highly motivated to work based on the approved goals and objectives.
9. Mistakes are viewed as learning capabilities and aid in re-examining processes and procedures.
10. The enterprise/organization is future-focused and adapts quickly to changing demands. People can articulate common goals and are aware when organizational goals are achieved.
11. Communication is frequent, both formal and informal, interactive, and multi-directional.
12. Strategies are data driven. The data is collectively analyzed and strategies and operational plans are developed from what is learned. There is an on-going cycle of gathering, analyzing, and making changes as needed.

3.0 Implementation

Once you have the above information, you can develop the overall Implementation Plan and Timeline and execute the processes and activities based on your documentation and policies. The Systems Engineering Capability Implementation Plan is a management tool designed to illustrate, in detail, the critical steps in implementing, using, maintaining and improving your

Systems Engineering capabilities. It is a guide or map that helps staff be proactive rather than reactive in accomplishing and using Systems Engineering and identifying any challenges along the way. It allows any person working in systems engineering (and all other concerned personnel), regardless of his or her level of involvement, to fully understand the goals and objectives outlined above and how they are to be accomplished. It ensures that everyone working on the program is on the same page and any discrepancies are resolved before they become costly to the program or population served. If you are accomplishing an enterprise or organizational Implementation Plan, this ensures that enterprise/organizational policies mandate consistency in how programs address and use Systems Engineering.

3.1 Implementation Plan

The first time frame is all about implementation planning and actual implementation in the enterprise. The next time frame should be about pilot testing each of the processes and activities as laid out in the Implementation Road Map and Time Line and tweaking implementation of the milestones as necessary. What elements are to be included in the Implementation Plan?

1. Implementation and Operational Time Line (especially if going to implement one process/activity at a time)
2. Necessary documentation and policies
3. Necessary training for existing personnel
4. Necessary skills and experience to hire
5. Automated metrics tools and necessary training times
6. Review periods for implementation progress versus original timeline
7. Review periods for Culture Change Management – Management Champions
8. Required changes to communication methodology (included in the overall timeline)
9. List of Processes/Metrics/Culture Change/Skill Sets/Communication Channels review periods

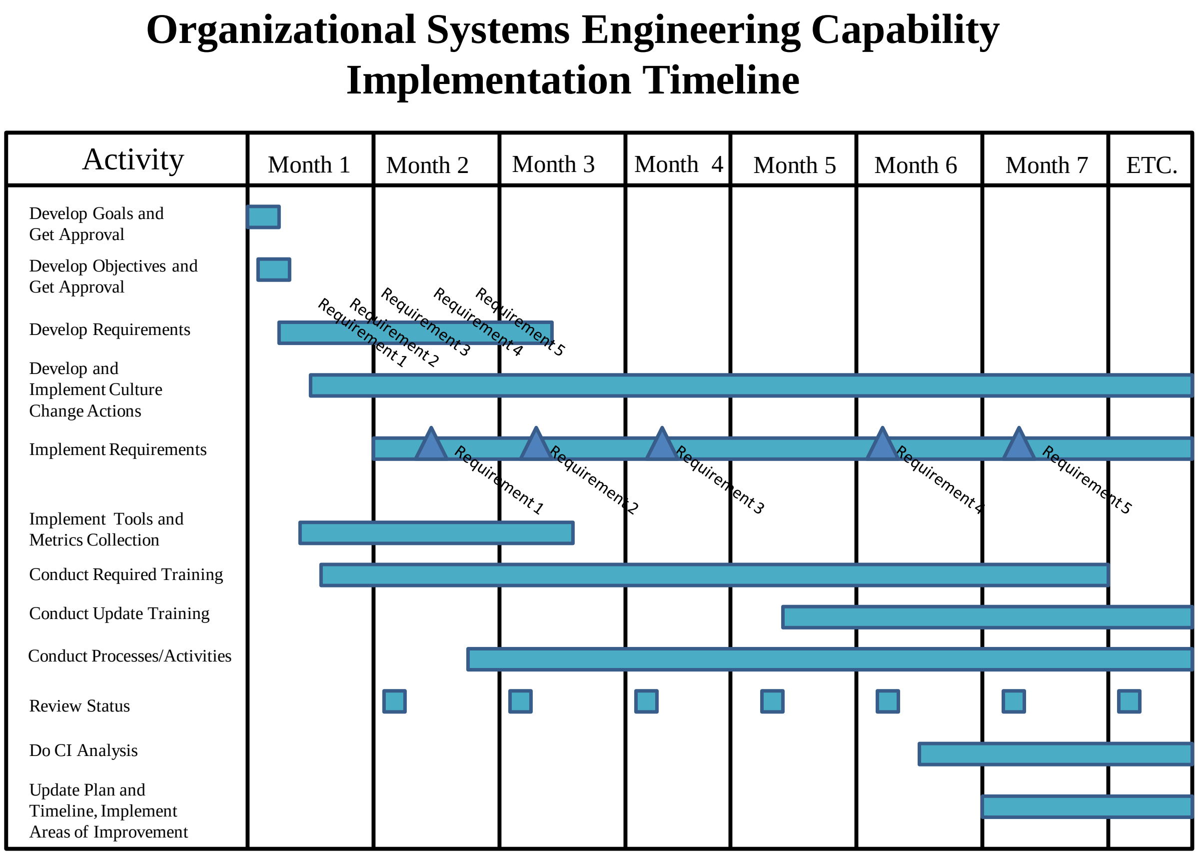


Figure 5: Example SE Capability Implementation Timeline

4.0 Operation and Sustainment

Once you have fully implemented one Systems Engineering process or activity[14], then all appropriate personnel should begin using that process or activity based on your documentation, policies and training. This assumes that you have implemented all necessary policy changes, metrics collection tools, process tools and communications strategy. It also assumes that your enterprise/organization/program have implemented labor collection tools that enable you to break out original work efforts and redo[15] work efforts separately.

[14] It is recommended that each process/activity be implemented individually according to an approved timeline. This type of implementation will minimize changes your personnel are required to accept and eliminate mass confusion.

[15] This is defined as work required to fix errors. It does not include work required by a change in requirements.

4.1 Review Each Process/Activity Periodically

Once your enterprise/organization/program has fully implemented a process or activity and is consistently using it, the process or activity should be reviewed (as scheduled in your Implementation timeline) to ensure that it is being carried out appropriately. These reviews should be carried out process by process or activity by activity unless one process/activity consistently feeds into another one. For example, Configuration Management and Baseline Control (see Requirements Development section above). These reviews should be simple to accomplish if the appropriate metrics and labor costs are being collected. All reviews should be documented and approved by senior management[16].

The following reviews should be accomplished periodically:

a) Review Metrics, Metrics Collection and Metrics Tools

b) Review Culture Change Progress

c) Review Communication Methodology

d) Review each process/activity requirement to ensure appropriate standards/best practices are being used

e) Review training requirements and personnel records to ensure appropriate training is being conducted

4.2 Personnel Evaluation and Enforcement

One question that plagues all existing implementations of Systems Engineering is "How do you know that your personnel are actually using the process/activity as required?" It is recommended that there be some evaluation and enforcement procedure established in enterprise/organizational/program policies and disseminated to all personnel. This procedure should be addressed in and become part of periodic performance appraisals. The metrics that are required for each requirement should be used in conjunction with labor metrics (original work versus redo work) to monitor (again periodically, not on an everyday basis) to establish how personnel are reacting to the required culture changes and what might be necessary for your Management Champions to do to ease the changes.

[16] During these reviews, you should also determine each requirement's validation/verification accuracy

5.0 Measurement

To accomplish the above, each company/organization/program must establish and use a measurement process (established metrics) that delivers relevant information to managers who use it for decision-making. Measurement information helps the manager(s) to:

- Monitor the progress and performance of the Systems Engineering process as well as the individual processes and activities
- Communicate effectively throughout the organization or company
- Identify and correct problems early (Continuous improvement, See Paragraph 6.0)
- Make key tradeoffs that affect how the Systems Engineering process is being used
- Track specific project objectives (the SE Goals and Objectives)
- Defend and justify decisions

For each of the above, measurement quantifies the relevant individual Systems Engineering processes or work products as well as the overall Systems Engineering Process with respect to the needs and objectives of the program, company or organization. Common system engineering metrics (see Section 2 Metrics List for suggested metrics) include timeliness, efficiency and effectiveness, performance requirements, quality attributes, conformance to standards and resource use. These measurements should also provide critical insight needed for continuous process improvement to achieve cost and schedule/cycle time reduction and quality and technical performance improvement.

Note that these metrics **must be** related to the validation and verification of the requirements established early in this process. They must allow each requirement to be verified and validated throughout the life cycle of the product(s).

6.0 Continuous Improvement

To support continuous improvement in the Systems Engineering Process as well as for individual SE processes and activities, all enterprises/organizations/programs should continually examine their processes to discover and eliminate problems. Typically, if the original goals/objectives/ requirements actions were accomplished successfully, this can be accomplished by making small changes within each process/activity rather than implementing any large-scale alteration. By focusing on making things better without finding blame, teams take actions to reduce errors, minimize defects and remove activities which provide no value and improve customer satisfaction. The Continuous Improvement (CI) processes referenced in this Guide (see reference section) feature a systems approach to improving the work flow in an enterprise/organization/program. Typical phases of a CI process are an analysis phase to identify specific problems, a design phase to determine what to do to remedy the problem(s), an implementation phase where the necessary actions are taken and an evaluation phase to monitor the outcome and determine if the adjustment to the process/activity has produced the desired result.

The following steps should be accomplished[17] and documented for each requirement regardless of the CI process chosen:

1. Review established goals and objectives to determine if any changes are required
2. Review each requirement and associated metrics

---

[17] Each requirement should be reviewed at a minimum once a year (Quarterly is recommended).

3. Determine what rework (not work related to requirements changes) is still ongoing and why
4. Review risks status and identify any problems/new risks/changes in risk assessment and level
5. Review culture change status and results
6. Update Implementation Plan and Time Line as required

7.0 Summary

Systems Engineering, when done optimally for your product(s), can significantly enhance your capabilities, enable your programs/projects to be completed on time and on budget with all required functions. But we note that this is not happening. In 2004, the Director for Systems Engineering in the Office of the Undersecretary for Defense for Acquisition, Technology and Logistics (OUSD [AT&L]) came to the National Defense Industrial Association (NDIA) and voiced concerns that DOD acquisition programs were not capitalizing on the value of systems engineering. He knew the value of SE and knew that it could help DOD programs, but he also knew that not all DOD program managers shared his convictions. Consequently program managers were taking shortcuts and eliminating SE capabilities from their programs. Subsequently, others have recognized this same problem. A recent Government Accountability Office (GAO) report indicates that acquisition program costs are typically 26 percent over budget and development costs are typically 40 percent more than initial estimates. These programs routinely fail to deliver the capabilities when promised, experiencing, on average, a 21 month delay. The report finds that "optimistic assumptions about system requirements, technology, and design maturity play a large part in these failures, and that these optimistic assumptions are largely the result of a lack of disciplined SE analysis early in the program."[18] This conundrum has continued to the present. The combined cost overrun for Major Defense Acquisition Program (MDAP) portfolio programs in 2015 was $468 billion, up from $295 billion in 2008. The total cost of the US Department of Defense's 2015 MDAP portfolio grew at 48.3

[18] From The Value of Systems Engineering posted on May 20, 2013 by Joseph Elm in Systems Engineering, https://insights.sei.cmu.edu/sei_blog/2013/05/the-value-of-systems-engineering.html

percent with an average schedule delay of 29.5 months.[19] For the latest statistics, do an Internet search on project management statistics.

The reason for this is that a Systems Engineering capability is normally defined and shaped by the context or environment in which it is embedded. But very seldom (or never) is the Systems Engineering Process used to define the optimum set of SE processes and activities necessary to provide maximum return on your investment in your environment. As discussed at the beginning of this book, companies and enterprises ad hoc embed individual SE processes and fail to determine the necessary interactions or the effectiveness of each process. Goals and objectives for the overall SE capabilities are not considered. So while some ROI is achieved, the maximum benefit of using appropriate, effective and efficient SE is not. Accomplishing implementation and use of Systems Engineering as described in this book can significantly increase the likelihood of successful program/project accomplishment and allow maximum return on investment.

[19] From Deloitte Aerospace Defense Report., 2017

Section II:
Systems Engineering Guidebook
Appendices

Systems Engineering Guidebook Section II

This Systems Engineering Guidebook is a template describing "**What"** to do to determine how to successfully implement and use appropriate (effective and efficient) Systems Engineering processes and activities. This section contains all of the Appendices pertaining to the Guidebook as noted in the Table of Contents. The objective of this Guide and the Appendices is to ensure that the Systems Engineering process meets the needs of the Enterprise/Organization/Program/Project while being scaled to the level of rigor that allows the system life cycle activities to be performed with an acceptable level of risk. **All** SE processes and activities must be tailored to a rigorous application that provides an appropriate level based on need. While all SE processes and activities apply to all life cycle stages, tailoring determines the process/activity level that applies to each stage, **and that level is never zero**. There is always some effort in each process and activity in each stage. At the enterprise or organizational level, the tailoring process adapts external standards in the context of the enterprise or organizational processes to meet the needs of the enterprise/ organization. At the program level, the tailoring process should adapt enterprise or organizational SE processes and activities to the unique needs of the program.

Appendix A: Required Expertise of SEIPT Personnel

It is recommended that, if you develop a Systems Engineering Integrated Process Team, a charter be established delineating the personnel roles, responsibilities and authority. The charter should also delineate the selected Senior Management champions and outline each role especially in the culture change procedure. Since Systems engineering is a problem-solving process used to translate operational needs and/or requirements into a well-engineered system solution. It is an interdisciplinary approach, including not only engineers, technical specialists, and customers, but also business and financial analysts. Systems engineering creates and verifies an integrated and life-cycle balanced set of system product, activity and process solutions that satisfy stated customer needs.

Systems Thinking

The approach of Systems Thinking is fundamentally different from traditional analysis. Traditional analysis focuses on separating the individual pieces of what is being studied/developed. Systems Thinking, in contrast, focuses on how the thing (component/part/ interface/etc.) being studied/developed interacts with the other constituents of the overall system – a set of elements that interact to produce behavior – of which it is a part. Systems Thinking also focuses on how the system being studied/developed interacts with the other systems as a part of a System of Systems. This means that instead of isolating smaller and smaller parts of the system being studied/developed, systems thinking works by expanding its view to take into account larger and larger numbers of interactions as an issue. This results in sometimes strikingly different conclusions that those generated by traditional analysis, especially when what is being studied/developed is dynamically complex or has a great deal of feedback from other internal or external sources.

Systems Engineering Expertise

1. Capability, domain or enterprise level engineering expertise.
2. Experience in technical management of products with similar capabilities.
3. Experience in interface engineering.

Breadth of Knowledge

1. Experienced in or capable of Systems Thinking (See below)
2. Knowledge across technical disciplines and engineering functions
3. Proven ability to ensure rigorous technical processes are applied
4. Experience in applying engineering capabilities, tools and techniques to anticipate issues with requirements, acquisition, test and sustainment of product capabilities.

Life Cycle Perspective

1. Experience in applying systematic processes and activities, specific technical processes and measurements that promote capability assurance throughout the product’s life cycle.

2. Capability to understand the Technical View[20], System View[21], Operational View[22] and Disposal View[23] of the product.
3. Experience in scope/range of requirements development, science and technology, product/system life cycle phases.
4. Experience in or knowledge of operational safety, suitability and effectiveness (OSS&E) characteristics and design.

The equivalent breadth and depth of knowledge as well as the overall vision should be required of all non-engineering SEIPT members.

[20] Technical View Success criteria – Systems/subsystem components function properly; designs reflect “plug and Play” open interfaces and domain industry standards.

[21] Systems View Success Criteria – Robust product and all subsystems function properly; product can operate and deliver required capability in its intended operational environment.

[22] Operational View – All required products can interoperate and potential operational errors are minimized.

[23] Disposal View – All aspects of the product should include an anticipated phase-out period and take disposal into account in the design and life cycle cost assessment.

Appendix B: Metrics

Cost Metrics		**When Collected**	**Where Briefed**	
1	Cost (Planned vs. Actual)	Monthly	Program Reviews	
2	Systems Cost (Estimate vs. Actual)	Monthly	Program Reviews	
3	Effort (Planned/Estimated vs. Actual)	Monthly	Program Reviews	
4	Margin	Monthly	Program Reviews	
5	Management Reserve Balance	Monthly	Program Reviews	
6	Estimate at Completion (EAC)	Weekly	Systems Engineering Reviews	
7	Estimate to Complete (ETC)	Weekly	Systems Engineering Reviews	
8	Planned/Estimated Labor Hours per Activity (BCWS)	Weekly	Systems Engineering Reviews	
9	Actual Labor Hours per Activity (BCWP)	Weekly	Systems Engineering Reviews	
10	Budgeted Cost of Work Performed (labor hours) per Activity (BCWP)	Weekly	Systems Engineering Reviews	
11	Variance at Completion (VAC)	Weekly	Systems Engineering Reviews	
12	Estimated Cost vs. Actual Cost per Activity	Weekly	Systems Engineering	

			Reviews	
13	Estimate vs. Actual Cost per Subsystem	Monthly	Systems Engineering Reviews	
14	Original Work vs. Rework Hours	Monthly	Program Reviews	
Staffing				
1	Planned vs Actual Staffing	Monthly	Program Reviews	
2	Planned vs. Actual Staffing Mix (Salary, Hourly, contractual)	Monthly	Program Reviews	
3	Planned vs. Actual Staffing Profile per Labor Category	Monthly	Program Reviews	
4	Unplanned Staff Losses per labor Category	Monthly	Program Reviews	
5	Unplanned Staff Gains per Labor Category	Monthly	Program Reviews	
Schedule				
1	Estimated vs. Actual IMS Schedule Milestones	Monthly	Program Reviews	
2	Risk Level at Each Schedule Milestone	Monthly	Program Reviews	
3	Estimated vs. Actuals Deliverables (Total, Complete, Remaining, Late)	Monthly	Program Reviews	
4	Deliverables (Aging)	Monthly	Program Reviews	
5	Critical Path status (Drag and Drag Cost)	Monthly	Program Reviews	
Quality				
1	On Time vs. Late Deliverables	Monthly	Program	

			Reviews	
2	Accepted vs. Rejected Deliverables	Monthly	Program Reviews	
3	Estimated vs. Actual Document Delivery Schedule Milestones	Monthly	Program Reviews	
4	Deliverables (Aging)	Monthly	Program Reviews	
Engineering Change Proposals				
1	Number of Customer Submitted ECPs by Month	Monthly	Program Reviews	
2	Estimated vs. Actual Dates Proposed, Open, Approved, Incorporated for Customer Submitted ECPs	Monthly	Program Reviews	
3	Number of Subsystems Affected by Customer Submitted ECPs	Monthly	Program Reviews	
4	Number of Computer Software Units (Program) Affected by Customer Submitted ECPs	Monthly	Program Reviews	
5	Number of CSU Lines of Code Affected by Customer Submitted ECPs	Monthly	Program Reviews	
6	Number of Documents/Drawings Affected by Customer Submitted ECPs	Monthly	Program Reviews	
7	Number of Contractor Submitted ECPs by Month	Monthly	Program Reviews	
8	Estimated vs. Actual Dates Proposed, Open, Approved, Incorporated Contractor Submitted ECPs	Monthly	Program Reviews	
9	Number of Subsystems Affected by Contractor Submitted ECPs	Monthly	Program Reviews	

10	Number of Computer Software Units (Program) Affected by Contractor Submitted ECPs	Monthly	Program Reviews	
11	Number of CSU Lines of Code Affected by Contractor Submitted ECPs	Monthly	Program Reviews	
12	Number of Documents/Drawings Affected by Contractor Submitted ECPs	Monthly	Program Reviews	
Failure Board Items				
1	Number of Failure/Incident Reports by Month	Weekly	Systems Engineering Review	
2	Number of Failure/Incident Reports Opened vs. Closed	Weekly	Systems Engineering Review	
3	Status of All Open Failure/Incident Reports	Weekly	Systems Engineering Review	
4	Assigned Risk Level for Each Failure/Incident Report	Weekly	Systems Engineering Review	
5	Estimated vs. Actual Cost Impact of each Failure/Incident report	Weekly	Systems Engineering Review	
6	Categorization of Failure/Incident Reports (Weight, cost, reliability, process, etc.)	Weekly	Systems Engineering Review	
7	Failure/Incident Reports Aging	Weekly	Systems Engineering Review	
Issues				
1	Number of Issues Established by	Weekly	Systems Engineering	

	Week		Review	
2	Categorization of Issues Established	Weekly	Systems Engineering Review	
3	Assigned Issue Risk	Weekly	Systems Engineering Review	
4	Number of Issues open vs. Closed	Weekly	Systems Engineering Review	
5	Estimated vs. Actual Cost Impact of each issue	Weekly	Systems Engineering Review	
6	Estimated vs. Actual Schedule impact of each issue	Weekly	Systems Engineering Review	
7	Issues Aging	Weekly	Systems Engineering Review	
Waivers/Deviations				
1	Number of Waivers/Deviations to Procedures/Processes/Parts by Week	Weekly	Systems Engineering Review	
2	Categorization of Waivers/Deviations to Procedures/Processes/Parts	Weekly	Systems Engineering Review	
3	Assigned Waivers/Deviations to Procedures/Processes/Parts Risk	Weekly	Systems Engineering Review	
4	Number of Waivers/Deviations to Procedures/Processes/Parts open vs. Closed	Weekly	Systems Engineering Review	
5	Estimated vs. Actual Cost Impact of each Waivers/ Deviations to	Weekly	Systems Engineering	

	Procedures/ Processes/Parts		Review	
6	Estimated vs. Actual Schedule impact of each Waivers/ Deviation to Procedures/ Processes/Parts	Weekly	Systems Engineering Review	
7	Waivers/Deviations to Procedures/ Processes/Parts Aging	Weekly	Systems Engineering Review	
Action Items				
1	Number of action Items Established by Week	Weekly	Systems Engineering Review	
2	Categorization of Action Items	Weekly	Systems Engineering Review	
3	Assigned Action Item Risk	Weekly	Systems Engineering Review	
4	Number of Action Items open vs Closed	Weekly	Systems Engineering Review	
5	Estimated vs. Actual Cost Impact of each action Item	Weekly	Systems Engineering Review	
6	Estimated vs. Actual Schedule impact of each Action Item	Weekly	Systems Engineering Review	
7	Action Item Aging	Weekly	Systems Engineering Review	
Risk				
1	History of Baseline Changes (including rationale for changes)			

2	Number of Risks Identified by Month	Monthly	Program Reviews	
3	Categorization of Risks (weight, size, cost, reliability, process, staffing, quality, schedule, etc.)	Monthly	Program Reviews	
4	Status of Identified Risks (pending, approved, rejected, closed, open, under assessment)	Monthly	Program Reviews	
5	Estimated vs. Actual Status of Each Risk Control Plan	Monthly	Program Reviews	
6	Value of each Identified Risk (Cost vs. Estimated Impact Cost, Schedule vs. Estimated Schedule Impact)	Monthly	Program Reviews	
7	Estimate vs. Actual Impact of Each Risk	Monthly	Program Reviews	
8	Number Open vs. Closed Risks	Monthly	Program Reviews	
9	Risk Aging	Monthly	Program Reviews	
Process Management				
1	History of Baseline Changes (including rationale for changes)			
2	Number of Systems Engineering Processes Documented/ Updated	Monthly	Program Reviews	
3	Planned vs. Actual Systems Engineering Processes Started	Monthly	Program Reviews	
4	Planned vs. Actual Systems Engineering Processes Operational	Monthly	Program Reviews	
5	Effectiveness of Process - Trend	Monthly	Program Reviews	
6	Process Compliance - Number	Monthly	Program	

	and Type of Escapes		Reviews	
7	Process Compliance Risk	Monthly	Program Reviews	
8	Continuous Improvement Audits by Process	Monthly	Program Reviews	
Configuration Management				
1	Number (Number of HWCIs and CSCIs)	Monthly	Program Reviews	
2	History of Baseline Changes (including rationale for changes)	Monthly	Program Reviews	
3	Change Requests - Number and Type	Monthly	Program Reviews	
4	Status of Change Requests (Submitted, Pending, Approved, Implemented)	Monthly	Program Reviews	
5	Cost Impact of Change Request (Estimated vs. Actual)	Monthly	Program Reviews	
6	Schedule Impact of Change Request (Estimated vs. Actual)	Monthly	Program Reviews	
7	Overall Cost and Schedule Impact of Change Requests (history of system)	Monthly	Program Reviews	
8	Change Request Risk Level	Monthly	Program Reviews	
9	Change Request Aging	Monthly	Program Reviews	
10	Number and Type of Deficiencies Identified by Configuration Audits	Monthly	Program Reviews	
11	Number of Document Configuration Items	Monthly	Program Reviews	
12	Status of Document Configuration Items (Open,	Monthly	Program Reviews	

	Draft, Final, Approved, Distributed)			
Environmental, Safety and Occupational Health (ESOH)				
1	Number of ESOH Requirements	Monthly	Program Reviews	
2	Number and Type of ESOH Functions and Constraints	Monthly	Program Reviews	
3	EOSH Performance Attributes	Monthly	Program Reviews	
4	Number of ESOH Hazards Identified and Acceptance Status	Monthly	Program Reviews	
5	Status of ESOH Hazard Action Plans	Monthly	Program Reviews	
6	Cost Impact of EOSH Plan (Estimated vs. Actual)	Monthly	Program Reviews	
7	Schedule Impact of ESOH Plan (Estimated vs. Actual)	Monthly	Program Reviews	
8	Overall Cost and Schedule Impact of ESOH Work Efforts (history of system)	Monthly	Program Reviews	
9	ESOH Plan Risk Level	Monthly	Program Reviews	
10	ESOH Hazards Aging	Monthly	Program Reviews	
Systems Engineering Training				
1	Personnel Experience Utilization (Trained and Qualified) by Category	Monthly	Program Reviews	

2	Number and Type of Training Hours Required per Year	Monthly	Program Reviews	
3	Number of Training Hours per Year by Employee (Required and Accomplished)	Monthly	Program Reviews	
4	Number of Training Courses (Required, In Progress, Developed, Provided)	Monthly	Program Reviews	
5	Training Evaluation and Feedback	Monthly	Program Reviews	
Manufacturing and Producibility				
1	Number and Type of Parts Manufactured/Modified	Monthly	Program Reviews	
2	Number and Type of Parts Purchased	Monthly	Program Reviews	
3	Parts Risk Status	Monthly	Program Reviews	
4	Estimated vs. Actual Production Time per Unit	Monthly	Program Reviews	
5	Total Labor Hours per Unit (Production, Inspection, Shipping, Installation, Maintenance, Removal)	Monthly	Program Reviews	
6	Number of Defects or Errors per unit	Monthly	Program Reviews	
7	Number and Type of Waivers/Deviations Requested/Approved per Unit	Monthly	Program Reviews	
8	Estimated vs. Actual Cost and Schedule Impact of Waiver/Deviation	Monthly	Program Reviews	
Technical Performance		Monthly	Program	

Measures			Reviews	
1	Estimated vs. Actual Reliability (system and subsystems)	Monthly	Program Reviews	
2	Estimated vs. Actual Operational Availability	Monthly	Program Reviews	
3	Estimated vs. Actual Maintainability	Monthly	Program Reviews	
4	Estimated vs. Actual Weight (systems and subsystems)	Monthly	Program Reviews	
5	Estimated vs. Actual Transportability	Monthly	Program Reviews	
6	Estimated vs. Actual Range	Monthly	Program Reviews	
7	Estimated vs. Actual Specific Fuel Consumption	Monthly	Program Reviews	
8	Others as Required by Product	Monthly	Program Reviews	
Customer Satisfaction				
1	Customer Survey/Questionnaire Results	Monthly	Program Reviews	
2	Number and Type of Customer Problem Reports	Monthly	Program Reviews	
3	Customer Reporting	Monthly	Program Reviews	
Lessons Learned				
1	Number of Lessons Learned Submitted	Monthly	Program Reviews	
2	Number of Lessons Learned Approved	Monthly	Program Reviews	
3	Number of Personnel Submitting	Monthly	Program	

	Lessons Learned		Reviews	
4	Number of Periodic Reviews of Lessons Learned	Monthly	Program Reviews	
5	Number of Personnel Reviewing Lessons Learned	Monthly	Program Reviews	
Software		Monthly	Program Reviews	
1	Estimate vs. Actual - Number of Unique Programs	Monthly	Program Reviews	
2	Estimate vs. Actual - Labor Hours per Program	Monthly	Program Reviews	
3	Estimate vs. Actual - Program Development Schedule	Monthly	Program Reviews	
4	Number and Type of User Complaints/Trouble Reports - System and Programs	Monthly	Program Reviews	
5	Time Required to Solve Complaints/Trouble Reports	Monthly	Program Reviews	
6	Number and Type of User Issue Reports - System and Programs	Monthly	Program Reviews	
7	Time Required to Solve User Issues	Monthly	Program Reviews	
8	Estimated vs. Actual Mean Time Between System Failures	Monthly	Program Reviews	
9	Number and Type of Programming Errors Found	Monthly	Program Reviews	
10	Time Required to Correct Programming Errors	Monthly	Program Reviews	
Parts, Materials and Processes				
1	Number and Type of DMSMS Issues	Monthly	Program Reviews	

2	Status of DMSMS Issues	Monthly	Program Reviews	
3	Risk of DMSMS Issues	Monthly	Program Reviews	
4	Estimated vs. Actual Cost Impact of each DMSMS Issue	Monthly	Program Reviews	
5	Estimated vs. Actual Schedule Impact of each DMSMS Issue	Monthly	Program Reviews	
6	Estimated vs. Actual Time Required to Successfully Address DMSMS Issues	Monthly	Program Reviews	
7	Number and Type of Counterfeit Parts Issues	Monthly	Program Reviews	
8	Status of Counterfeit Parts Issues	Monthly	Program Reviews	
9	Risk of Counterfeit Parts Issues	Monthly	Program Reviews	
10	Estimated vs. Actual Cost Impact of each Counterfeit Parts Issue	Monthly	Program Reviews	
11	Estimated vs. Actual Schedule Impact of each Counterfeit Parts Issue	Monthly	Program Reviews	
12	Estimated vs. Actual Time Required to Successfully Address Counterfeit Parts Issues	Monthly	Program Reviews	
13	Number of Parts Reviewed	Monthly	Program Reviews	
Hardware				
1	Estimate vs. Actual - Number of Unique Components	Monthly	Program Reviews	
2	Estimate vs. Actual - Labor Hours per Component	Monthly	Program Reviews	
3	Estimate vs. Actual -Component	Monthly	Program	

	Development Schedule		Reviews	
4	Number and Type of User Complaints/Trouble Reports - System and Components	Monthly	Program Reviews	
5	Time Required to Solve Complaints/Trouble Reports	Monthly	Program Reviews	
6	Number and Type of User Issue Reports - System and Components	Monthly	Program Reviews	
7	Time Required to Solve User Issues	Monthly	Program Reviews	
8	Estimated vs. Actual Mean Time Between System Failures	Monthly	Program Reviews	
9	Number and Type of Design Errors Found	Monthly	Program Reviews	
10	Time Required to Correct Design Errors	Monthly	Program Reviews	
11	Design Margin - components, subsystems, system	Monthly	Program Reviews	
Requirements				
1	Total Number of Requirements	Monthly	Program Reviews	
2	Total Number of Requirements by Tier (if appropriate)	Monthly	Program Reviews	
3	Number of Non-Compliances and reason	Monthly	Program Reviews	
4	Risk by Requirement and reason	Monthly	Program Reviews	
5	Requirements Verified versus Requirements Not Verified	Monthly	Program Reviews	
6	Requirements Validated versus Requirements Not Validated	Monthly	Program Reviews	

7	Number of Requirements Changed and reason	Monthly	Program Reviews	
8	Number of Requirements Added and reason	Monthly	Program Reviews	
9	Total Number of Requirements Allocated by Requirement	Monthly	Program Reviews	
10	Number of Non-Compliances in Allocated Requirements and reason	Monthly	Program Reviews	
11	Risk by Allocated Requirement and reason	Monthly	Program Reviews	
12	Allocated Requirements Verified versus Requirements Not Verified	Monthly	Program Reviews	
13	Allocated Requirements Validated versus Requirements Not Validated	Monthly	Program Reviews	
14	Number of Allocated Requirements Changed (after original baseline established) and reason	Monthly	Program Reviews	
15	Number of Allocated Requirements Added (after original baseline established) and reason	Monthly	Program Reviews	
16	Number of Interface Requirements Established	Monthly	Program Reviews	
17	Number of Interface Requirements Non-Compliances and reasons	Monthly	Program Reviews	
18	Risk by Interface Requirement and reason	Monthly	Program Reviews	
19	Interface Requirements Verified versus Requirements Not Verified and reason	Monthly	Program Reviews	

20	Interface Requirements Validated versus Requirements Not Validated and reason	Monthly	Program Reviews	
21	Number of Interface Requirements Changed (after original baseline established) and reason	Monthly	Program Reviews	
22	Number of Interface Requirements Added (after original baseline established) and reason	Monthly	Program Reviews	
Technical Planning				
1	Title/Type of Technical Plans Required and Budgeted versus Completed and Approved	Monthly	Program Reviews	
2	Status of Technical Plan Development - Estimated versus Actual by Plan (possible to use EVM if planning is via a WBS level)	Monthly	Program Reviews	
3	Periodic Review Date and Amount of Updating/Revising (by Plan)	Monthly	Program Reviews	
4	Timeline and status of Milestones for Temporal relationships, Hierarchical relationships, Critical Dependencies relationships, and Supporting Infrastructure relationships by Plan	Monthly	Program Reviews	
Technical Assessment Effort Metrics				
1	Technical Performance Measures (TPM) derived from Key Performance Parameters (KPPs)	Monthly	Program Reviews	

	and Key System Attributes (KSAs)			
2	Technical progress (at both the system and system element levels)	Monthly	Program Reviews	
3	Software metrics (e.g., size, complexity, reuse, defects, productivity)	Monthly	Program Reviews	
4	Hardware metrics (space, weight and power (SWaP), processing margin, axle loading, available RAM, etc.)	Monthly	Program Reviews	
5	Technical staffing	Monthly	Program Reviews	
6	Technology maturity	Monthly	Program Reviews	
7	Affordability	Monthly	Program Reviews	
8	Risk Mitigation	Monthly	Program Reviews	
9	Schedule	Monthly	Program Reviews	
10	Quality/manufacturing/production measures (e.g., defects, first pass yields, process escapes)	Monthly	Program Reviews	
11	Infrastructure measures (e.g., capacity, availability, utilization of facilities and equipment)	Monthly	Program Reviews	
12	Design/development process measures (e.g., drawing releases, software modules, subsystem integration tasks, defined/documented interfaces, deviations, waivers, etc.)	Monthly	Program Reviews	
Validation and				

Verification				
	Methods of Verification/Validation	**Pass – Fail Criteria based on:**	**When Collected**	**Where Briefed**
1		Incorporation of markups of deficiencies from all approvers	Monthly	Program Reviews
2	Peer Reviews (software)	formal peer reviews, informal contacts and a separate testing group.	Monthly	Program Reviews
3	Peer Reviews (drawings)	Incorporation of markups of deficiencies from all approvers. Signatures	Monthly	Program Reviews
4	Peer Reviews (technical publications)	Incorporation of markups of deficiencies from editorial	Monthly	Program Reviews
5	Peer Reviews (kits, drawings)	Incorporation of markups of deficiencies from all approvers	Monthly	Program Reviews
6	Peer Reviews (packaging data)	Incorporation of all editors' mark-ups of deficiencies	Monthly	Program Reviews
7	Customer Validation	Incorporation of markups from Hands-	Monthly	Program Revie

		on Validation (w/wo customer)		ws
8	Customer Verification	Incorporation of markups from Hands-on Verification (w/customer)	Monthly	Program Reviews
		Defined Test Procedure(s) IAW ASTM D4169 and MIL-STD-2073 appendix F	Monthly	Program Reviews
9	Testing, Packaging	Any Test Procedure(s) provided by the customer	Monthly	Program Reviews
10	Testing, parts	Defined in Test Procedure	Monthly	Program Reviews
11	Testing, vehicles	Defined in Test Procedure	Monthly	Program Reviews
12	Testing, software	Defined in Test Procedure	Monthly	Program Reviews
13	Inspection (in process)	Components have passed when no open issues are indicated on deficiency	Monthly	Program Reviews

		sheet(s)		
14	Inspection (receiving)	Components have passed when no open issues are indicated on Inspection Test Report (ITR)	Monthly	Program Reviews
15	Safety Inspection	Components have passed when no open issues are indicated on the deficiency sheet(s).	Monthly	Program Reviews
16	Safety Inspection, Software	Defined in Safety Test Procedure	Monthly	Program Reviews
		Defined Test Procedure(s) IAW ASTM D4169 and MIL-STD-2073 appendix F	Monthly	Program Reviews
17	Final Inspection	Defined in Software Safety Test Procedure	Monthly	Program Reviews
		Components have passed when no open issues are indicated on the deficiency sheet(s).	Monthly	Program Reviews

Software Metrics				
1	Balanced scorecard	Monthly	Program Reviews	
2	Bugs per line of code	Monthly	Program Reviews	
3	Code coverage	Monthly	Program Reviews	
4	Cohesion	Monthly	Program Reviews	
5	Comment density[1]	Monthly	Program Reviews	
6	Connascent software components	Monthly	Program Reviews	
7	Coupling	Monthly	Program Reviews	
8	Cyclomatic complexity (McCabe's complexity)	Monthly	Program Reviews	
9	DSQI (design structure quality index)	Monthly	Program Reviews	
10	Function Points and Automated Function Points	Monthly	Program Reviews	
11	Halstead Complexity	Monthly	Program Reviews	
12	Instruction path length	Monthly	Program Reviews	
13	Maintainability index	Monthly	Program Reviews	
14	Number of classes and interfaces	Monthly	Program Reviews	
15	Number of lines of code	Monthly	Program Reviews	
16	Number of lines of customer requirements	Monthly	Program Reviews	

17	Program execution time	Monthly	Program Reviews	
18	Program load time	Monthly	Program Reviews	
19	Program size (binary)	Monthly	Program Reviews	
20	Weighted Micro Function Points	Monthly	Program Reviews	
21	CISQ automated quality characteristics measures	Monthly	Program Reviews	
22	defect arrival and fix rate (used to can help measure the maturity of the code)	Monthly	Program Reviews	
Systems Integration				
1	Technical integration strategy document Status,	Monthly	Program Reviews	
2	Integration Plans Status	Monthly	Program Reviews	
3	Integration test scripts status	Monthly	Program Reviews	
4	Integration test scenarios status (both development and implementation)	Monthly	Program Reviews	
5	Integration tests status and results (include any retests and reason for retest)	Monthly	Program Reviews	

Appendix C: References

These are a few of the references I thought might be useful when researching how to accomplish each step. Many more can be reached through searching for the phrase of each area. I recommend that each step, if never accomplished before, be researched thoroughly to enable you to determine the most effective way to accomplish each step.

Overall

1. INCOSE Systems Engineering Handbook, A Guide for Systems Engineering Processes and Activities, Fourth Edition, INCOSE-TP-002-04, 2015
2. Guide to the Systems Engineering Body of Knowledge (SEBoK), v1.6, March 2016, http://sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK)
3. NASA Systems Engineering Handbook, NASA/SP-2007-6105 Rev 1
4. MITRE Systems Engineering Guide, 2014, http://www.mitre.org/sites/default/files/publications/se-guide-book-interactive.pdf
5. Space and Missile Center Systems Engineering Primer and Handbook: Concepts, Processes, and Techniques, Space & Missile Systems Center, U.S. Air Force, 3rd Edition 29 April 2005
6. NATIONAL AIRSPACE SYSTEM (NAS): SYSTEM ENGINEERING MANUAL (SEM) - FEDERAL AVIATION ADMINISTRATION (FAA) (VER. 3.1) (11 OCT 2006) [S/S BY FAA SEM 1.0.1] (NAS Systems Engineering Portal (SEP), https://sep.faa.gov/)

Architecture/Design References

1. Guide To Running Software Development Projects, IBM, 2003http://www.ibm.com/developerworks/websphere/library/techarticles/0306_perks/perks.html
2. System **Design**& **Technical Architecture** https://www2.gov.bc.ca/assets/gov/british-columbians-our-governments/services-policies-for-government/information-technology/standards/economy-sector/system_design__technical_architecture_template.docx

Baseline Control

1. https://vca.berkeley.edu/sites/default/files/change_control_process_aa.pdf
2. MIL-HDBK-61 page Page 3-4, "Configuration baseline (baseline)"
3. https://www.energy.gov/projectmanagement/downloads/evms-training-snippet-46-baseline-control-methods

Culture Measurement and Change

1. Culture as Culprit: Four Steps to Effective Change, http://executiveeducation.wharton.upenn.edu/thought-leadership/wharton-at-work/2011/09/four-steps-culture-change
2. How to Change Your Company Culture, By Susan M. Heathfield, October 12, 2016, https://www.thebalance.com/how-to-change-your-culture-1918810
3. Understanding the Culture of your Organization, (adapted from the work of Edgar H. Schein, Sloan Fellows Professor of Management Emeritus), 2016
4. The project champion: key to implementation success: https://www.pmi.org/learning/library/project-champion-key-implementation-success-2135

5. In Change Management, Start With Champions, Not Antagonistshttps://www.forbes.com/sites/markmurphy/2015/06/25/in-change-management-start-with-champions-not-antagonists/#61772f4ebd0a
6. The 12 Attributes of a Strong Organizational Culture, By Charles Rogel March 18, 2014, https://www.tlnt.com/the-12-attributes-of-a-strong-organizational-culture/
7. 7 Benefits Of Mistake-Driven Learning, Christopher Pappas April 10, 2015 https://elearningindustry.com/7-benefits-of-mistake-driven-learning

Measurement

1. INCOSE Systems Engineering Measurement Primer v2.0, Document No.: INCOSE-TP-2010-005-02, 5 November 2010
2. ISO/IEC 15939:2007, Systems and software engineering — Measurement Process, Software Engineering Institute (SEI) CMMI®-DEV and CMMI®-ACQ — Measurement and Analysis process area
3. Practical Software and Systems Measurement (PSM) ISO/IEC 15288:2008, Systems and software engineering — System life cycle processes
4. INCOSE-TP-2005-003-02, Systems Engineering Leading Indicators Guide, version 2.0, dated 29 January 2010
5. INCOSE-TP-2003-020-01, Technical Measurement: A Collaborative Project of PSM, INCOSE, and Industry

Qualification, Validation and Verification Standards

ISO has a range of standards for quality management systems that are based on ISO 9001 and adapted to specific sectors and industries. These include:

1. ISO/TS 16949 – Automotive production and relevant service part organizations
2. ISO/TS 29001 – Petroleum, petrochemical and natural gas industries
3. ISO 13485 – Medical devices
4. ISO/IEC 90003 – Software engineering
5. ISO 17582 – Electoral organizations at all levels of government
6. ISO 18091 - Local government
7. AS 9100, Quality Systems - Aerospace - Model for Quality Assurance in Design, Development, Production, Installation and Servicing, 1999-11-01
8. IEEE Standard 1012 2004, IEEE Standard for Software Verification and Validation; IEEE Computer Society

Risk Management References and Standards

1. ISO 31000:2009, Risk management – Principles and guidelines, provides principles, framework and a process for managing risk. It can be used by any organization regardless of its size, activity or sector. Using ISO 31000 can help organizations increase the likelihood of achieving objectives, improve the identification of opportunities and threats and effectively allocate and use resources for risk treatment. However, ISO 31000 cannot be used for certification purposes, but does provide guidance for internal or external audit programs. Organizations using it can compare their risk management practices with an internationally recognized benchmark, providing sound principles for effective management and corporate governance

2. System of Systems Engineering Collaborators Information Exchange (SoSECIE) http://www.acq.osd.mil/se/outreach/sosecollab.html
3. ISO Guide 73:2009, Risk management - Vocabulary complements ISO 31000 by providing a collection of terms and definitions relating to the management of risk.
4. ISO/IEC 31010:2009, Risk management – Risk assessment techniques focuses on risk assessment. Risk assessment helps decision makers understand the risks that could affect the achievement of objectives as well as the adequacy of the controls already in place.
5. ISO/IEC 31010:2009 Risk Management - Risk Assessment Techniques focuses on risk assessment concepts, processes and the selection of risk assessment techniques.
6. A Risk Management Standard – IRM/Alarm/AIRMIC 2002 – developed in 2002 by the UK's 3 main risk organizations.
7. OCEG "Red Book" 2.0: 2009 - a Governance, Risk and Compliance Capability Model
8. BS 31100: 2008, Code of Practice for Risk Management
9. COSO: 2004, Enterprise Risk Management - Integrated Framework
10. FERMA: 2002, A Risk Management Standard
11. SOLVEN CY II: 2012, Risk Management for the Insurance Industry
12. Department of Defense Risk, Issue, and Opportunity Management Guide for Defense Acquisition Programs, June 2015, Office of the Deputy Assistant Secretary of Defense for Systems Engineering

Systems Thinking

1. Simple_Complexity: A Management Book for the Rest Of Us – A Guide to Systems Thinking, William Donaldson, 2017, Morgan James Publishing
2. http://www.systemicleadershipinstitute.org/systemic-leadership/theories/basic-principles-of-systems-thinking-as-applied-to-management-and-leadership-2/
3. Systems Thinking, Systems Tools and Chaos Theory, http://managementhelp.org/systems/index.htm
4. Human Factors in Simple and Complex Systems, R.W.Proctor and T.Van Zandt, CRC Press, 2008

Systems Engineering Integrated Product Team

1. Systems Engineering Working Level Integrated Product Team: www.acq.osd.mil/se/docs/SE-WIPT-Generic-Charter-Template.doc
2. Integrated Product Team – https//en.wikipedia.org/wiki/Integrated-product-team
3. Integrated Product Team Start-Up Guide, MITRE, https://www.mitre.org/publications/technical-papers/integrated-project-team-ipt-startup-guide
4. DoD Integrated Product and Process Development Handbook, August 1998, OFFICE OF THE UNDER SECRETARY OF DEFENSE (ACQUISITION AND TECHNOLOGY)
5. Rules of the Road: A Guide for Leading Successful Integrated Product Teams. Rules of the Road can be found at http://www.acq.osd.mil/ar/ipt.htm..

Systems Engineering Processes and Activities

1. Defense Acquisition Guidebook, 2016, https://acc.dau.mil/CommunityBrowser.aspx?id=289207&lang=en-US

2. Systems engineering, From Wikipedia, the free encyclopedia, https://en.wikipedia.org/wiki/Systems_engineering
3. INCOSE Systems Engineering Handbook, Version 4, 2016
4. Guide to the Systems Engineering Body of Knowledge (SEBoK)
5. Naval Systems Engineering Guidebook, NAVAL SYSTEMS ENGINEERING STEERING GROUP WASHINGTON DC, ADA527494, OCT 2004
6. NASA Systems Engineering Handbook (NASA/SP-2007-6105/Rev1), December 2007
7. Systems Engineering Guide for Systems of Systems, Version 1.0, August 2008, Director, Systems and Software Engineering Deputy Under Secretary of Defense (Acquisition and Technology) Office of the Under Secretary of Defense (Acquisition, Technology and Logistics)
8. Systems Engineering Standards and Models Compared, Sarah Sheard and J. Lake, https://www.researchgate.net/publication/228556143_Systems_Engineering_Standards_And_Models_Compared
9. Seven Systems Engineering Myths and the Corresponding Realities, J. Kasser, Proceedings of the Systems Engineering Test and Evaluation Conference, Australia, 2010
10. The HKM Framework for Systems Engineering, J. Kasser, INCOSE International Symposium, 2007.

Systems Engineering Training

1. http://appel.nasa.gov/developmental-programs/seldp/nasa-systems-engineering-training-courses/
2. Training Offered by Government Agencies, http://www.nist.gov/standardsgov/training.cfm
3. INCOSE Systems Engineering Handbook, Version 3.2a. INCOSE. 2012.
4. Systems Engineering Fundamentals. Defense Acquisition University Press, 2001
5. NASA Systems Engineering Handbook. NASA/SP-2007-6105 Rev1, December 2007.
6. US DoD Systems Management College Systems Engineering Fundamentals, Defense Acquisition University Press, 2001
7. Guide to the Systems Engineering Body of Knowledge (SEBoK), INCOSE, 2010
8. Joint Software Systems Safety Engineering Handbook, Version 1.0 Published August 27, 2010
9. IEEE 1220 – 2005 Standard for Application and Management of the Systems Engineering Process, IEEE Press, 2009
10. ISO/IEC 15288 – 2008 Systems and Software Engineering System Life Cycle Processes
11. ISO 12207 Software Engineering and Development
12. ANSI/EIA 632 Processes for Engineering a System
13. MIL-STD-499A Engineering Management
14. Risk Management Guide for DOD Acquisition, Fifth Edition, Version 2.0, 2003
15. MIL-STD-882E DoD Standard Practice for System Safety, 2012

Systems Integration

1. International Standards for System Integration, Richard A. Martin, 2005, ISO TC184/SC5/WG1 ISO/TC 184/SC 5 - Interoperability, integration, and architectures for enterprise systems and automation applications
2. NASA/SP-2010-3407, The Human Integration Design Handbook (HIDH)
3. NASA-STD-3000, the Man-System Integration Standards (Superseded and is no longer being maintained but is an excellent reference)
4. Best Practices for Systems Integration, Copyright © 2011 Northrop Grumman Systems Corporation. All rights reserved. Log #DSD-11-78

Systems Engineering "Standards"

1. MIL-STD-499 Series Covers Systems Engineering Management
2. ANSI/EIA 632 (1999) Covers processes for engineering a system
3. IEEE 1220 (1998) Standard for the application and management of the Systems Engineering Process
4. ISO/IEC 15288 (2002) Lists processes performed by Systems Engineers – applicable to the role of Systems Engineers rather than the activities known as Systems Engineering. Many activities overlap those of project management.
5. Systems Engineering Standards: A Summary, http://www.acqnotes.com/Attachments/Systems%20Engineering%20Standards%20A%20Summary.pdf

Technical Planning Process

1. US Army Corps of Engineers, Engineer Manual 200-1-2, TECHNICAL PROJECT PLANNING (TPP) PROCESS (Download from http://www.usace.army.mil/inet/usace-docs/eng-manuals/em.htm.)
2. DEFENSE ACQUISITION GUIDEBOOK, Chapter 4 -- Systems Engineering, Section 4.3.2. Technical Planning Process (https://acc.dau.mil/CommunityBrowser.aspx?id=638326)
3. NASA NPR 7123.1B, Appendix C. Practices for Common Technical Processes

Tailoring Standards and References

1. ISO 15288 tailoring focuses on the deletion of unnecessary or unwarranted process elements, but it does allow for additions and modifications as well.
2. ISO/IEC TR 24748-1 (2010)
3. ISO/IEC TR 24748-2 (2010)
4. Tailoring Systems Engineering Projects for Small Satellite Missions, Stephen Horan and Keith Belvin, NASA Langley Research Center AVT-210 / RSM-031
5. Langley Research Center, LaRC NPR7120.5D and NPR7123.1A Tailoring Guidelines (Baseline 1), http://engineering.larc.nasa.gov/
6. Tailoring Systems Engineering Processes for Integration of Research and Prototyping Activities, ISESS Andrew Tokmakoff , Alex Farkas, Sam Mosel, DOI Bookmark: http://doi.ieeecomputersociety.org/10.1109/SESS.1999.766575
7. Tailoring Systems Engineering Lifecycle Processes to Meet the Challenges of Project and Programme Applications, Richard David Adcock BSc MSCINCOSE International Symposium, Volume 15, Issue 1, Version of Record online: 4 NOV 2014

8. Tailoring a Large Organization's Systems Engineering Process to Meet Project-Specific Needs, Matthew Graviss, Shahram Sarkani, and Thomas A. Mazzuchi, Defense ARJ, http://dau.dodlive.mil/2016/06/22/tailoring-a-large-organizations-systems-engineering-process-to-meet-project-specific-needs/

Configuration Management References and Related Standards

(list from http://cmpic.com/configuration-management-standards.htm)

1. Aerospace
   - AS9100C - Advanced Quality System
   - DO 178B - Software Considerations in Airborne Systems and Equipment
   - ATIS 0300250, Operations, Administration, Management & Provisioning (OAM&P) — Extension to Generic Network Model for Interfaces between Operations Systems and Network Elements to Support Configuration Management—Analog and Narrowband ISDN Customer Service Provisioning. Alliance for Telecommunications Industry Solutions (ATIS)
   - American National Standards Institute (ANSI) ANSI INCITS TR-47, Information Technology—Fiber Channel-Simplified Configuration and Management Specification (FC-SCM)

2. American Nuclear Society (ANS)
   - ANSI/ANS-3.2-1982, Administrative Controls & Quality Assurance for the Operational Phase of Nuclear Power Plants

3. American Society of Mechanical Engineers (ASME)
   - ASME/NQA-1, Quality Assurance Program Requirements for Nuclear Facilities
   - ASME/NQA-1B-2011, Quality Assurance Requirements for Nuclear Facilities Applications
   - ASME/NQA-2A-1990, Addenda to ASME NQA-2-1989 Edition, Quality Assurance Requirements for Nuclear Facility Applications
   - ANSI/N45.2.11, Quality Assurance Requirements for the Design of Nuclear Power Plants
   - ANSI/N18.7, Administrative Controls & Quality Assurance for the Operational Phase of Nuclear Power Plants
   - ANSI/N45.2.9-1974, Requirements for the Collection, Storage, & Maintenance of Quality Assurance Records for Nuclear Power Plants
   - ANSI/N45.2.13, QA Requirements for the Control of Procurement of Items & Services for Nuclear Power Plants
   - ASME Y14.100, Government/Industry Drawing Practices
   - ASME Y14.24M Types & Applications of Engineering Drawings
   - ASME Y14.34M, Parts Lists, Data Lists & Index Lists
   - ASME Y14.35M, Revision of Engineering Drawings & Associated Documents
   - ASME NQA Committee, Task Group - CM Draft, January 1991

4. Australia
   - AS/NZS 3907:1996, Quality Management—Guidelines for Configuration Management

- Australian Army Configuration Management Manual (CMMAN) Army CM Manual; version 3.0
- DI(G) LOG 08-4, Configuration Management of Systems and Equipment
- DI(A) SUP 24-2, Configuration Management Policy within Army [currently being updated for issue as DI(A) LOG XX-X, Configuration Management Policy for Capabilities in the Land Environment]
- DI(A) LOG 1-33, Integrated Logistic Support and the Army Material Process (currently being updated)
- The Army Specification Manual (SPECMAN)
- EMEI Workshop A 850, Modifications, Trial Modifications, and Local Modifications to Equipment
- MINE WARFARE-STD-499B, Systems Engineering

5. Automotive Industry Action Group (AIAG) (North American Automotive Industry)
   - QS9000, Quality System Requirements (replaced by ISO/TS16949)
   - ISO/TS 16949 "Quality management systems particular requirements for the application of ISO 9001:2000 for automotive production and relevant service part organizations"

6. American Society for Quality (ASQ)
   - ANSI/ASQ Q9004-1994, Quality Management & Quality System Elements- Guidelines for CM Draft Standard for Facilities, 5/8/93; never finished?

7. British Standards Institute (BSI)
   - BS 6488, CM of Computer Based Systems
   - BSI PD ISO/IEC TR 18018, Information Technology—Systems and Software Engineering—guide for configuration management tool capabilities
   - BS EN 46001-Application of EN 29001 (BS5750: Part 1) to the manufacture of medical devices
   - BS5515:1984- British Code of Practice for Documentation of Computer Based Systems
   - BS 7799- Information Security Management
   - BS 15000-1 IT Service Managament defines the requirements for an organization to deliver managed services of an acceptable quality for its customers. Note: replaced by ISO 20000-1?
   - BS 15000-2 IT Service Managament best practices for Service Management processes. Note: replaced by ISO 20000-2?
   - BSI PAS 55:2008 "Asset Management"
   - BS EN 13290-5:2001, Space Project Management. General requirements configuration management

8. Canada - Department of National Defense (DND) Standards
   - C-05-002-001/AG-00, Aerospace Engineering Change Proposal Procedures
   - D-01-000-200/SF-001, Joint Electronics Type Designation System (JETS)
   - D-01-002-007/SG-001, Requirements for the Preparation of CM Plans
   - D-01-002-007/SG-002, Requirements for Configuration Identification
   - D-01-002-007/SG-004, Requirements for Configuration Status Accounting

- D-01-002-007/SG-006, Requirements for the Selection of Configuration Items
- D-01-100-215/SF-000, Specification for Preparation of Material Change Notices
- D-01-400-001/SG-000, Engineering Drawing Practices
- D-01-400-002/SF-000, Drawings, Engineering & Associated Lists
- D-02-002-001/SG-001, Identification Marking of Canadian Military Property
- D-02-006-008/SG-001, The Design Change, Deviation & Waiver Procedure

9. Chrysler / Ford / General Motors
   - QS-9000, Quality System Requirements

10. Code of Federal Regulations (CFR) - United States and other Governments
    - Clinger-Cohen Act (IT)
    - Sarbanes-Oxley, "Define and establish controls..." is the heart of the Sarbanes Oxley Act
    - Title 21 CFR Part 820, Quality System Medical Devices FDA
    - Title 10 CFR Part 830 122 , Quality Assurance Criteria DOE
    - Title 14 CFR Chapter I (FAA), Part 21 Certification Procedures for Products and Parts
    - Title 48 CFR, Federal Acquisition Regulations
    - Title 48 CFR 2210 Specifications, Standards and Other Purchase Descriptions
    - Title 48 CFR 1 Part 46 Quality Assurance

11. Commercial U.S. Airlines Ait Transportation Association (ATA)
    - ATA 100
    - ATA 200
    - ATA 2100
    - ATA 2200
    - In 2000, ATA Spec 100 and ATA Spec 2100 were incorporated into ATA iSpec 2200: Information Standards for Aviation Maintenance. ATA Spec 100 and Spec 2100 will not be updated beyond the 1999 revision level.

12. Department of Defense (DoD) (United States)

DoD Superseded / Canceled:

- AMCR 11-12, Total Decision Making Process
- AMCR 11-26, CM, (Army,1965)
- ANA Bulletin NO. 390, Engineering Change Proposal (Army, Navy, Air Force)
- ANA Bulletin 391, Engineering Change Proposal
- ANA Bulletin 445, Engineering Changes to Weapons, Systems, Equipments, & Facilities, (1963)
- AR 70-37, Joint DOD Service Agency Regulation Configuration Management
- AFSCM 375-1, CM During the Development & Acquisition Phases, (Air Force Systems Command, 1962)
- AFSCM 375-3, System Management (1964)
- AFSCM 375-4, System Program Management Procedures (1966)
- AFSCM 375 -5, Systems Engineering Management Procedures (1966)
- AFSCM 375-7, Configuration Management for Systems, Equipment, Munitions, and Computer Programs (1971)

- AFWAMAN33-2 AIR FORCE WEATHER AGENCY CONSOLIDATED NETWORK CONFIGURATION MANAGEMENT PLAN (2003)
- AMC INSTRUCTION 33-105 ENTERPRISE CONFIGURATION MANAGEMENT
- ASWSPO 5200.4 (Navy, 1965))
- Army Regulation 25-6, Configuration Management for Automated Information Systems
- BuWeps Instruction 5200.20, Processing Engineering Change Proposals (NAVY)
- CMI, CM Instructions, Air Force Systems Command, Space Systems Division (1963):
- CMI No. 1, Facility Engineering Change Proposal procedures (1964)
- CMI No. 2, Engineering Change Proposal Procedures (1964)
- CMI No. 3, Specification Maintenance (1964)
- CMI No. 4, Configuration Change Implementation (1964)
- CMI No. 5, Configuration Accounting Procedures (1964)
- CMI No. 7, Configuration Control Board (1964)
- CMI No. 9, First Article Configuration Inspection (1964)
- DOD 5000.2-M, Defense Manual- Defense Acquisition Management Documents & Reports
- DOD 5000.19, Policies for the Management & Control of Information Requirements
- DOD 5010.12, Management of Technical Data (Superseded by online ASSIST database, https://assist.dia.mil/online/start/index.cfm)
- DOD 5010.19, DOD CM Program
- DOD 5010.21, CM Implementation Guidance
- DOD 8000 series, Policies & Procedures for Automated Information Systems
- DOD-D-1000- Drawing, Engineering & Associated Lists
- DOD-STD-2167, Defense System Software Development
- DOD-STD-7935, DOD Automated Information System Documentation Standards
- M200 (1962), Standardization Policies, Procedures & Instructions, Defense Standardization Manual, then replaced by 4120.3-M in 1966
- DOD-HDBK-287, A Tailoring Guide For DOD-STD-2167A
- MIL-STD-12, Abbreviations for Use on Drawings, Specs, Standards, & Technical Documents (will eventually be replaced by ANSI Y14.38
- MIL-STD-100, Engineering Drawing Practices (Superseded by ASME-Y14.100, ASME-Y14.24, ASME-Y14.35M and ASME-Y14.34M)
- MIL-STD-454, Standard General Requirements for Electronic Equipment
- MIL-STD-480B,Configuration Control- Engineering Changes, Deviations & Waivers
- MIL-STD-481, Configuration Control- Short Form
- MIL-STD-482, Configuration Status Accounting Data Elements & Related Features
- MIL-STD-483, CM Practices for Systems, Equipment, Munitions, & Computer Programs
- MIL-STD-490, Specification Practices
- MIL-STD-498, Software Design & Development (replaces Dod-STD- 2167, DOD- STD-7935, & DOD-STD-1703)
- MIL-STD-499, Systems Engineering
- MIL-STD-999, Certification of CM/DM Process (DRAFT)
- MIL-STD-1456, CM Plan
- MIL-STD-1521, Technical Reviews & Audits for Systems, Equipments, & Computer

- Software (Appendixes G, H, & I were superseded by MIL-STD-973)
- MIL-STD-1679, Software Development (1978)
- MIL-STD-2549, Configuration Status Accounting (canceled 9/3/0/2000). However, ARMY is using AMC-STD-2549A until EIA 836 is published
- MIL-STD-3046, DOD Interim Standard Practice Configuration Management (Army, 2013)
- MIL-D-70327, Drawings & Data Lists
- MIL-Q-9858, Quality Program Requirement
- MIL-S-52779, Software Quality program Requirements (1974)
- MIL-STD-31000A, Technical Data Packages
- NAVAIRINST 4000.15, Management of Technical Data & Information
- NAVMAT INSTR 4130.1, (Navy, 1967)
- NAVMATINSTR 4131.1, CM, (Navy, 1967)
- COMDTINST 4130.6A, Coast Guard Configuration Management Policy
- COMDTINST M4130.8, Coast Guard Configuration Management for Acquisitions and Major Modifications
- COMDTINST M4130.9, Coast Guard Configuration Management for Sustainment
- NAVMATINSTR 5000.6, CM, (Navy, 1966)
- NAVSEA 0900-LP-080-2010, Software Configuration Control Procedures Manual, (NAVY, 1975)
- (U)- E-759/ESD, Software QA Plan, (Air Force. 1980)
- OPNAVINST 4130.1.; Configuration Management of Software in Surface Ship Combat Systems, (Navy. 1975)

13. Department of Defense (DoD) (United States)
Active -many have been canceled but are active in existing contracts, some have been replaced with commercial versions. Check the government web sites for latest status.

- US COE EC 11-2-173, USCCE Manpower Civil Program Civilian Air Force Configuration
- and Management
- US COE ER 15-1-33, Automation Configuration Management Boards
- AFPD 21-4 Engineering Data
- AFI 21-401 Engineering Data Storage, Distribution, & Control
- AFI 21-402 Engineering Drawing System
- AFI 21-403 Acquiring Engineering Data
- AFMCPAM63-104 IWSM Configuration Management Implementation Guide (2000)
- AFWAMAN33-2 Air Force Weather Agency Consolidated Network Configuration Management Plan (2003)
- AMCI33-105 Configuration Management 15 Oct 2000
- DID Guide, HQ AFMC/EN DID Guide
- DOD 5000.1, Defense Acquisition
- DOD 5000.2, Defense Acquisition Management Policies & Procedures
- DoD 5010.12-M, Procedure for the Acquistion and Management of Technical Data
- DoD 5010.12-L, Acquisition Management System & Data Requirement List (AMSDL)

- DoD Cataloging Handbook H6, Federal Item Identification Guides for Supply Cataloging
- DoD Cataloging Handbook H7, Manufacturers Part & Drawing Numbering Systems for Use in the Federal Cataloging System
- DoDISS, Department of Defense Index of Specifications & Standards)
- MIL-HDBK-59- Computer Aided Acquisition & Logistics Support (CALS) Program Implementation Guide (CALS is now known as Continuous Acquisition & Life Cycle Support)
- MIL-HDBK-61, Configuration Management
- MIL-STD-109, Quality Assurance terms & Definitions
- MIL-STD-1168, Lot Numbering of Ammunition
- MIL-STD-130, Identification Marking of US Military Property
- MIL-HDBK-245, Preparation of Statement of Work
- MIL-STD-280, Definition of Item levels, Item Exchangeability, Models & RelatedTerms
- MIL-HDBK-454, Standard General Requirements for Electronic Equipment
- MIL-STD-881, Work Breakdown Structure for Defense Material Items
- MIL-STD-961, Military Specifications & Associated Documents, Preparation of
- MIL-STD-962D, Defense Standards Format and Context.
- MIL-STD-963C, Data Item Descriptions (DIDs)
- MIL-STD-973, CM Notice 3 (canceled 9/30/2000)
- MIL-STD-974, CITIS (Contractor Integrated Technical Information Service, is being transitioned to a non-government standard).
- MIL-STD-1309, Definitions of Terms for Test, Measurement & Diagnostic Equipment
- MIL-STD-1465, CM of Armaments, Munitions & Chemical Production Modernization
- MIL-STD- 1520, Corrective Action & Disposition System for Non Conforming Material
- DOD-STD-1700, Data Management Program (not superseded but generally replaced by SAE-GEIA-859, Data Management, 24 Nov 2014)
- MIL-STD-1767, Procedures for Quality assurance & Configuration Control of ICBM Weapon System Technical Publications & Data
- MIL-STD-1840, Automated Interchange of Technical Information
- MIL-STD-2084, General Requirements for Maintainability of Avionics & Electronic Systems & Equipment
- DOD-STD-2168, Defense System Software Quality Program
- MIL-I-8500, Interchangeability & Replaceability of Component Parts for Aerospace Vehicles
- MIL-S-83490, Specification, Types & Forms
- SMC-S-002, Configuration Management (Space and Missile Command)
- USAFAI33-114 Managing Software Configuration and Controlling Data in the Cadet
- Administrative Management Information System (CAMIS)

14. DoD - Draft Documents
    - MIL-STD-CNI, Coding, Numbering, & Identification
    - SD-15, Performance Specification Guide
    - AD-A278-102, Blueprint for Change (regarding use on commercial standards, obtain through NTIS)

15. Department of Energy (DOE) (United States)
NOTE: Check for recent status, as standards are being canceled and replaced on a regular basis. Also see www.ac-incorp.com/CM_Standards.html

- DOE Order 430.1 Life Cycle Asset Management
- DOE Guide G-830-120 Implementation Guide for 10CFR Part 830.120, QA
- DOE Order 4330.4A Maintenance Management Program
- DOE Order 4700.1 Project Management System (will be phased out)
- DOE Order 5480.19 Conduct of Operations Requirements for DOE Facilities
- DOE Order 5700.6C Quality Assurance
- DOE Order 6430.1A General Design Criteria
- DOE-STD-1073-93 Parts 1 & 2, Guide for Operational CM Program
- NPO 006-100 DOE Office of New Production Reactors CM Plan
- Code of Federal Regulations Title 10, Part 830 (Energy)
- DOE- never released
- DOE 5480.CM, Operational CM Program (see DOE-STD-1073)
- DOE Draft, no number, CM for Non-Nuclear Facilities

16. Department of Transportation, Federal Highway Administration (DOT/FHA) (United States)

- Configuration Management for Transportation Management Systems Handbook, FHWA-OP-04-013 http://ops.fhwa.dot.gov/freewaymgmt/publications/cm/handbook/index.htm
- Configuration Management Fact Sheet http://ops.fhwa.dot.gov/freewaymgmt/publications/cm/factsheet/
- Configuration Management Primer http://ops.fhwa.dot.gov/freewaymgmt/publications/cm/primer/
- Configuration Management Tri-Fold Brochure http://ops.fhwa.dot.gov/freewaymgmt/publications/cm/brochure/
- Configuration Management Technical Presentation http://ops.fhwa.dot.gov/freewaymgmt/publications/cm/presentation/index.htm

17. Electronic Industries of America (EIA)

- ANSI/EIA-632-1 Draft Process for Engineering a System- Part 1: Process Characteristics
- ANSI/EIA-632-2 Draft Process for Engineering a System- Part 2:Implementation Guidance
- ANSI/EIA-649B Configuration Management (now written by TechAmerica)
- CMB 3 Recommendations Concerning CM Audits
- CMB 4-1 CM Definitions for Digital Computer programs
- CMB 4-2 Configuration Identification for Digital Computer Programs
- CMB 4-3 Computer Software Libraries
- CMB 4-4 Configuration Change Control for Digital Computer Programs
- CMB 5 CM Requirements for Subcontractors/Vendors
- CMB 6-1 Configuration & Data Management References
- CMB 6-2 Configuration & Data Management In-House Training Plan
- CMB 6-3 Configuration Identification
- CMB 6-4 Configuration Control

- CMB 6-5 Textbook for Configuration Status Accounting
- CMB 6-6 Reviews & Configuration Audits
- CMB 6-7 Data Management
- CMB 6-8 Data Management In-House Training Course
- CMB 6-9 Configuration & Data Management Training Course
- CMB 6-10 Education in Configuration & Data management
- CMB 7-1 Electronic Interchange of CM Data
- CMB 7-2 Guideline for Transitioning CM to an Automated Environment
- CMB7-3 CALS CM SOW & CDRL Guidance
- EGSA 107 Glossary of DoD CM Terminology & Definitions
- EIA/IS 632 System Engineering
- EIA-748 Earned Value Management Systems
- EIA-927, Common Data Schema for Complex Systems
- EIA SP 3537 Processes for Engineering a System
- EIA SP 4202 On-Line Digital Information Service (ODIS)
- EIA SSP 3764 Standard for Information Technology- Software Life Cycle Processes Software Development Acquirer/ Supplier Agreement
- IEEE/EIA 12207.0 Industry Implementation of ISO/IEC 12207 (Standard for Information Technology)
- IEEE/EIA 12207.1 Guide for Information technology- Software Life Cycle Processes Life Cycle Data
- IEEE/EIA 12207.2 Guide for Information Technology- Software Life Cycle Processes Implementations Considerations
- J-Std-016 (EIA/IEEE Interim Standard) Standard for Information Technology; Software Life Cycle Processes; Software Development; Acquirer/Supplier Agreement
- Systems Engineering EIA-632
- Systems Engineering Capability Model SECM EIA-732

18. Electric Power Research Institute (EPRI)

Also see: http:www.ac-incorp.com/CM_standards.html

- EPRI TR-103586, Guidelines for Optimizing the Engineering Change process for Nuclear Power Plants, prepared by Cygna Energy services, Oakland, CA
- EPRI NP-5640, Nuclear Plant Modifications & Design Control: Guidelines for Generic Problem Prevention
- EPRI NP-6295, Guidelines for Quality Records in electronic Media for Nuclear Facilities
- EPRI NP-3434, Value-Impact Analysis of Selected Safety Modifications to Nuclear Power Plants
- EPRI NP- 5618, Enhancing Plant Effectiveness Through Improved Organizational Communication
- EPRI NSAC-121, Guidelines for Performing Safety System Functional Inspections

19. European Cooperation for Space Standardization (ECSS) http://www.ecss.nl/

- ECSS-M-ST-10, Space Project Management—Project Planning and Implementation
- ECSS-M-ST-40C, Space Project Management—Configuration and Information Management

- ECSS-Q-ST-10-09, Space Product Assurance—Nonconformance Control System
- ECSS-Q-ST-20, Space Product Assurance—Quality Assurance

20. European Community for Standardization (ECS)
    - JAR-21, Certification Procedures for Aircraft & Related Products & Parts (Draft)
    - CEN EN 13290-5, Space Project Management—General Requirements—Part 5: Configuration Management
    - EN 13290-6, Space Project Management—General Requirements—Part 6: Information/ Documentation Management
    - EN14160, Space Engineering—Software
    - EN 9200, Programme Management—Guidelines for Project Management Specification

21. European Computer Manufacturers Institute (ECMI)
    - ECMA-TR 47, CM Service Definition

22. European Defense Standards Reference System (EDSTAR)
    - CEN Workshop 10, European Handbook for Defense Procurement Expert Group 13 Life Cycle (Project) Management Final Report
    - ECSS-M-40A, Configuration Management
    - ECSS-E-10, System Engineering
    - ECSS-M-50, Information/Documentation Management
    - ECSS-E-40, Space System Software Engineering. http://www.eda.europa.eu/EDSTAR/home.aspx

23. European Space Agency (ESA)
    - Software Engineering Standards, ESA PSS-05-0
    - Guide to Software CM, ESA PSS-05-09, ISSN 0379-4059
    - Guide to Software Verification & Validation, ESA PSS-05-10
    - Guide to Software Quality Assurance, ESA PSS-05-11
    - European Computer Manufacturers Institute
    - ECMA-TR 47, CM Service Definition

24. European Telecommunications Standards Institute (ETSI)

Too many to list: http://www.etsi.org/standards

25. Federal Aviation Administration (FAA) (United States)
    - Title 14 Code of Federal Regulations, Parts 1-59
    - FAA-STD-002 Facilities Engineering Drawing Practices
    - FAA-STD-005 Preparation of Specification Documents
    - FAA-STD-018 Computer Software Quality Program (1977)
    - FAA-STD-021 CM Contractor Requirements
    - FAA-STD-058, Standard Practice Facility Configuration Management
    - FAA Order 1800.8 National Airspace Systems CM
    - FAA Order 6030.28 National Airspace Systems CM

- FAA 1100.57 National Engineering Field Support Division Maintenance Program Procedures, Operational Support (AOS)
- FAA 1800.63 National Airspace System (NAS) Deployment Readiness Review (DRR) Program
- FAA 1800.66 National Policy Configuration Management Requirements
- FAA 6032.1 Modifications to Ground Facilities, Systems, and Equipment in the NAS

26. Food and Drug Administration (FDA) (United States)
    - FDA 8541-79, Good Manufacturing Practices, Food & Drug Administration (superseded by QSR)
    - QSR Quality System Regulation for the Medical Device Industry (QSR) (based on ISO 9001)

27. Federal Highway Administration (FHA) (United States)
    - Configuration Management for Transportation Management Systems Handbook (FHWA Publication Number: FHWA-OP-04-013) (EDL Document Number: 13885)
    - A Guide to Configuration Management for Intelligent Transportation Systems, (FHWA
    - Publication Number: FHWA-OP-02-048) (EDL Document Number: 13622)
    - Configuration Management for Transportation Management Systems Primer (FHWA Publication Number: FHWA-OP-04-014) (EDL Document Number: 13886)
    - Configuration Management for Transportation Management Systems Brochure (FHWA Publication Number: FHWA-OP-04-016) (EDL Document Number: 13888)
    - Configuration Management for Transportation Management Systems Fact Sheet(FHWA Publication Number: FHWA-OP-04-017) (EDL Document Number: 13889)
    - Configuration Management for Transportation Management Systems Technical Presentation

28. France
    - NF EN 13290-5 January 2002, Management of Space Projects—General Requirements—Part 5: Configuration Management
    - AFNOR NF EN 300291-1, Telecommunications Management Network (TMN)—Functional Specification of Customer Administration (CA) on the Operations System/ Network Element (OS/NE) Interface—Part 1: Single Line Configurations (V1.2.1)
    - AFNOR NF ETS 300617, Digital Cellular Telecommunications System (Phase 2)—GSM Network Configuration Management

29. Germany Deutsches Institut für Normung (DIN)
    - DIN EN 13290-5, Aerospace–Space Project Management—General Requirements— Part 5: Configuration Management, German and English versions
    - DIN EN 300291-1, Telecommunications Management Network (TMN)—Functional Specification of Customer Administration (CA) on the Operations System/Network Element (OS/NE) Interface—Part 1: Single Line Configurations [Endorsement of the English version EN 300291-1 V 1.2.1 (1999–02) as the German standard]
    - DIN EN 300291-2, Telecommunications Management Network (TMN)—Functional Specification of Customer Administration (CA) on the Operations System/Network

Element (OS/NE) Interface—Part 2: Multiline Configurations [Endorsement of the English version EN 300291-2 V 1.1.1 (2002–03) as the German standard]
- DIN EN 300376-1, Telecommunications Management Network (TMN)—Q3 Interface at the Access Network (AN) for Configuration Management of V5 Interfaces and Associated User Ports—Part 1: Q3 Interface Specification [Endorsement of the English version EN 300376-1 V 1.2.1 (1999–10) as German standard]
- DIN ETS 300617, Digital cellular telecommunications system (Phase 2)—GSM Network Configuration Management; English version ETS 300617
- DIN EN 300377-1, Telecommunications Management Network (TMN)—Q3 Interface at the Local Exchange (LE) for Configuration Management of V5 Interfaces and Associated Customer Profiles—Part 1: Q3 Interface Specification [Endorsement of the English version EN 300377-1 V 1.2.1 (1999–10) as the German standard]
- DIN ETS 300377-2, Signaling Protocols and Switching (SPS)—Q3 Interface at the Local Exchange (LE) for Configuration Management of V5 Interfaces and Associated Customer Profiles—Part 2: Managed Object Conformance Statement (MOCS) Performance Specification; English version ETS 300377-2:1995
- DIN EN 300820-1, Telecommunications Management Network (TMN)—Asynchronous Transfer Mode (ATM) Management Information Model for the X Interface Between Operation Systems (OSs) of a Virtual Path (VP)/Virtual Channel (VC) Cross-Connected Network—Part 1: Configuration Management [Endorsement of the English version EN 300820-1 V 1.2.1 (2000-11) as the German standard]
- DIN EN 301268, Telecommunications Management Network (TMN)—Linear Multiplex Section Protection Configuration Information Model for the Network Element (NE) View [Endorsement of the English version EN 301268 V 1.1.1 (1999-05) as the German standard]
- VG 95031-1, Modification of Products—Part 1: Procedure According to CPM
- VG 95031-2, Drawing Set—Part 3: Parts List
- VG 95031-3, Drawing Set—Part 3: Changes on Drawings

30. Institute of Electrical & Electronic Engineers (IEEE)
    - IEEE 323, Qualifying Class IE Equipment for Nuclear Power Generating Stations
    - IEEE 344, Recommended Practices for Seismic Qualification of Class IE Equipment for Nuclear Power Generating Stations
    - IEEE 352, Guide for the General Principles of Reliability Analysis of Nuclear Power Generating Station Safety Systems
    - IEEE Std 610, (ANSI), Computer Dictionary
    - IEEE Std 730-1989, (ANSI), Software QA Plans
    - IEEE Std 828-90, (ANSI), Standard for Software CM Plans
    - IEEE Std 830-84, (ANSI), Guide for Software Requirements Specifications
    - IEEE Std 1028, (ANSI), Standard for Software Reviews & Audits
    - IEEE Std 1042 (ANSI), Guide for Software CM
    - IEEE Std 1063 (ANSI), Standard for User Documentation
    - IEEE Std P1220 (ANSI), Systems Engineering
    - IEEE Std 803-1983, Recommended Practice for Unique Identification in Power Plants & Related Facilities - Principles & Definitions

- IEEE Std 803.1-1992, Recommended Practice for Unique Identification in Power Plants & Related Facilities - Component Function Identifiers
- IEEE Std 804-1983, Recommended Practice for Implementation of Unique Identification System in Power Plants & Related Facilities
- IEEE Std 805-1984 , Recommended Practice for System Identification in Nuclear Power Plants & Related Facilities
- IEEE Std 806-1986, Recommended Practice for System Identification in Fossil-Fueled Power Plants & Related Facilities
- IEEE/EIA 12207.0 Industry Implementation of ISO/IEC 12207 (Standard for Information Technology)
- IEEE/EIA 12207.1 Guide for Information technology- Software Life Cycle Processes Life Cycle Data
- IEEE/EIA 12207.2 Guide for Information Technology- Software Life Cycle Processes Implementations Considerations
- J-Std-016 (EIA/IEEE Interim Standard) Standard for Information Technology; Software Life Cycle Processes; Software Development; Acquirer/Supplier Agreement
- IEEE 828-2012-IEEE Standard for Configuration Management in Systems and Software Engineering: Institute of Electrical and Electronics Engineers / 16-Mar-2012 / 71 pages

31. Institute of Nuclear Power Operations (INPO) (United States)

Also see: http://www.ac-incorp.com/CM_standards.html

- GP MA-304, Control of Vendor Manuals
- GP TS-402, Plant Modification Control Program
- GP TS-407, Computer Software Modification Controls
- GP TS-412, Temporary Modification Control
- GP TS-415, Technical Reviews of Design Changes
- GP TS-4l l, Temporary Lead Shielding
- INPO 85-016, Temporary Modification Control
- INPO 85-031, Guidelines for the Conduct of Technical Support Activities at Nuclear Power Stations
- INPO 86-006, Report on Configuration Management in the Nuclear Utility Industry
- INPO 87-006, Report on Configuration Management in the Nuclear Utility Industry
- INPO 87-006, Report on CM in the Nuclear Industry
- INPO 88-009, System & Component labeling
- INPO 88-016, Guidelines for the Conduct of Design Engineering
- INPO 90-009 REV. 1 Guidelines For The Conduct of Design Engineering
- INPO 94-003, A Review of Commercial Nuclear Power Industry Standardization Experience

32. International Organization for Standardization (ISO)

NOTE: Standards can be searched at http://www.iso.org/iso/home.html

- ISO 9000:2000 Series
- ISO 9000:2000, Quality management systems - Fundamentals and vocabulary
- Note: replaces ISO 9001, ISO 9002 and ISO 9003.
- ISO 9001:2000, Quality management systems - Requirements

- ISO 9004:2000, Quality management systems - Guidelines for performance improvements
- ISO 19011, Guidelines on Quality and/or Environmental Management Systems Auditing (under development)
- ISO 10005:1995, Quality management - Guidelines for quality plans
- ISO 10006:2003, Quality management systems - Guidelines for quality management in projects
- ISO 10007:2003 Guidelines for Configuration Management
- ISO/DIS 10012, Parts 1 and 2, Quality assurance requirements for measuring equipment
- ISO/DIS 10303-239 Industrial automation systems and integration -- Product data representation and exchange -- Part 239: Application protocol: Product life cycle support
- ISO 10013:1995, Guidelines for developing quality manuals
- ISO/TR 10014:1998, Guidelines for managing the economics of quality
- ISO 10015:1999, Quality management - Guidelines for training
- ISO/TS 16949:1999, Quality management systems particular requirements for the application of ISO 9001:2000 for automotive
- production and relevant service part organizations. Joint effort IAOB (International Automotive Oversight Bureau) and ISO
- ISO/IEC 17025 General requirements for the competence of testing and calibration laboratories
- ISO 8402 Quality Management & Quality Assurance Vocabulary
- ISO 9000-1 Guidelines for Use of the ISO 9000 Series (replaced by ISO 9000:2000)
- ISO 9000-2 Guidelines for Applying ISO 9000 to Services (replaced by ISO 9000:2000)
- ISO 9000-3 Guidelines for Applying ISO 9000 to Software (replaced by ISO 9000:2000?)
- Note: TickIT is an ISO 9000 accreditation scheme for software developers and supporters.
- ISO 9001 Model for Quality Assurance in Design/Development, Production, Installation, & Servicing. (replaced by ISO 9001:2000)
- ISO 9001:2008 Quality management systems - Requirements
- ISO/DIS 10303-239 Industrial automation systems and integration -- Product data representation and exchange -- Part 239: Application protocol: Product life cycle support
- ISO 9002 Model for Quality Assurance in Production & Installation. Note: replaced by ISO 9001:2000 (replaced by ISO 9001:2000)
- ISO 9003 Model for Quality Assurance in Final Inspection & Test. Note: replaced by ISO 9001:2000 (replaced by ISO 9001:2000)
- ISO 9004 Quality Management & Quality System Elements- Guidelines (replaced by ISO 9004:2000)
- ISO 9004-7 Guidelines for CM (Draft) never released under this number. It was released as ISO10007
- ISO 10011-1, -2 and -3 Guidelines for Auditing Quality Systems
- ISO 10303-1: 1994 Industrial Automation Systems and Integration -Product Data Representation and Exchange. This standard had too many parts to list here. NOTE: Check ISO for corrections and revisions in this 10303 series. The collection appears to be subject to many changes.

- ISO 12207 Information technology - Software life cycle processes
- ISO 13485 Quality systems - Medical devices - Particular requirements for the application of ISO 9001
- ISO 14001 Environmental management systems -Specification with guidance for use
- ISO 14004 Environmental management systems -General guidelines on principles, systems and supporting techniques
- ISO/IEC TR 15504-1 to -8:1998 Information technology - Software process assessment - Parts 1 through 8
- ISO20000-1: 2005 IT Service Management
- ISO20000-2: 2005 IT Service Management code of practice, describing specific best practices for the processes within ISO 20000-1.
- Also, there are industry specific variations of ISO 9000: see canceled QS9000 (automotive), TL9000 (Telecommunications), QSRs (FDA) and AS9000 (Aerospace)

33. International Versions
    - AENOR UNE-EN ISO 10007, Quality Management Systems—Guidelines for Configuration Management (Spanish)
    - AFNOR FD ISO 10007, Quality Management Systems—Guidelines for Configuration Management (French)
    - AENOR UNE-ISO 10007, Quality Management Systems—Guidelines for Configuration Management (Spanish)
    - UNI ISO 10007, Quality Management Systems—Guidelines for Configuration Management (Italian)
    - CSA CAN/CSA-ISO 10007:03, Quality Management Systems—Guidelines for Configuration Management (Canadian)
    - TSE TS EN ISO 10007, Quality Management Systems—Guidelines for Configuration Management (Turkish)
    - SNV SN EN ISO 10007, Quality Management Systems—Guidelines for Configuration Management (ISO 10007:1995); Trilingual version (English, German, and French)

34. International Telecommunications Union (ITU)
    - ITU-T Q.824.5, Stage 2 and Stage 3 Description for the Q3 Interface—Customer Administration: Configuration Management of V5 Interface Environment and Associated Customer Profiles—Series Q: Switching and Signaling Specifications of Signaling System
    - ITU-R S.1252, Network Management—Payload Configuration Object Class Definitions for Satellite System Network Elements Forming Part of SDG Transport Networks in the Fixed-Satellite Service
    - ITU-T J.705, IPTV Client Provisioning, Activation, Configuration and Management Interface Definition
    - ITU-T X.792, Configuration Audit Support Function for ITU-T Applications—Series X: Data Networks and Open System Communications OSI Management—Management Functions and ODMA Functions

35. National Aeronautics and Space Administration (NASA) (United States)
NOTE: Check for recent status, as standards are being canceled and replaced on a regular basis.
NASA - Superseded:

- NPC 500-1 (or NHB 8040.2), Apollo CM Manual (released 1964) & MSC Supplement #1 (1965)
- PC-093, Maintenance & Configuration Control Requirements, NASA Pioneer Program (1965)

NASA - Active:

- NASA GPR 1410.2, Configuration Management (GSFC)
- NASA-LLIS-2596, Lessons Learned—Management Principles Employed in Configuration Management and Control in the X-38 Program
- NASA MPR 8040.1, Configuration Management, MSFC Programs/Projects
- NASA MWI 8040.1, Configuration Management Plan, MSFC Programs/Projects
- NASA MWI 8040.7, Configuration Management Audits, MSFC Programs/Projects
- NASA-STD 0005, NASA Configuration Management Standard
- NASA Software Configuration Management Guidebook (1995) from Software Assurance Technology Center
- NASA-STD-2201-93, Software Assurance Standard www.hq.nasa.gov/office/codeq/doctree/ canceled/220193.pdf
- GMI 8040.1A CM (for satellite or ground system projects)
- JSC 30000 includes CM Requirements (for space station)
- JSC 31010 CM Requirements
- JSC 31043 CM Handbook
- KHB8040.2B CM Handbook
- KHB 8040.4 Payloads CM Handbook
- KPD 8040.6B CM Plan, National Space Transport System
- MM8040.5C CM Accounting & Reporting System
- MM 8040.12 Contractor CM Requirements
- MM8040.13A Change Integration & Tracking System
- MM1 8040.15 CM Objectives, Policies & Responsibilities
- MMI 8040.15B CM
- MSFC-PROC-1875 Contractor CM Plan Review Procedure
- MSFC-PROC-1916 CM Audit Procedures for MSFC Programs/Projects
- NSTS 07700, Volume IV, Configuration Requirements, Level II Program Definition & Requirements
- SSP 30000 Program Definition & Requirements Document
- Configuration Management Requirements, Space Station Project Office, October 29, 1990

36. National Institute of Standards and Technology (NIST) (U.S.)

- NIST 800-53 Recommended Security Controls for Federal Information Systems

37. North Atlantic Treaty Organization (NATO)

- ACMP-1 Requirements for Preparation of CM Plans

- ACMP-2 Requirements for Identification
- ACMP-3 Requirements for Configuration Control
- ACMP-4 Requirements for Configuration Status Accounting
- ACMP-5 Requirements for Configuration Audits
- ACMP-6 NATO CM Terms & Definitions
- ACMP-7 Guidance on Application of ACMPs 1-6
- ACMP-2009: (DRAFT) NATO Guidance on Configuration Management
- ACMP-2100: (DRAFT) NATO Contractual Configuration Management Requirements
- AQAP-1: NATO Requirements for an Industrial Quality Control System
- AQAP-13 Software Quality Control Requirements
- AQAP-1 NATO Requirements for an Industrial Quality Control System
- AQAP-160: NATO Integrated Quality Requirements for Software Throughout the Life Cycle
- AQAP-2110: Requirements for Design, Development, and Production
- STANAG 4159 NATO Material CM Policy & Procedures
- STANAG 4427 Introduction to Allied Configuration Management

38. Norway

Norwegian Defense Acquisition Regulation (ARF)

- ISO 10007 with a contractual adaption replaced in 2014 by NATO ACMP 2100 suppliers quality management systems shall comply with Allied Quality Assurance Requirements—AQAPs

39. Nuclear Information & Records Management Association (NIRMA)

Also see: http:www.ac-incorp.com/CM_standards.html

- CM 1.0 -2000 (R2006) DRAFT Configuration Management of Nuclear Facilities
- PP02-1989, Position paper on CM
- PP03-1992, Position Paper for Implementing a CM Enhancement Program for a Nuclear Facility
- PP04-1994, Position Paper for CM Information Systems
- TG14-1992, Support of Design Basis Information Needs
- TG19-1996, CM of Nuclear Facilities
- TG20-1996, Drawing Management-Principles & Processes

40. Nuclear Regulatory Commission (NRC) / NRC Regulatory Guides (U.S.)

- 1.28, QA Program Requirements (Design & Construction)
- 1.33, QA program Requirements (Operations)
- 1.33, Rev 2, QA Requirements for the Design of Nuclear Power Plants
- 1.64, Quality Assurance Requirements for the Design of Nuclear Power Plants
- 1.88, Collection, Storage, & maintenance of Power Plant Quality Assurance Records
- 1.123, QA Requirements for Control of Procurement of Items & Services for Nuclear Power Plants
- 1.152, Criteria for Programmable Digital Computer System Software in Safety- Related Systems of Nuclear Power Plants
- NUREG BR 0167, Software Quality Assurance Programs & Guidelines

- NUREG 1000, Generic Implications of ATWS Events at the Salem Nuclear Power Plant
- NUREG/CR 1397, An Assessment of Design Control Practices & Design Reconstitution Programs in the Nuclear Power Industry
- NUREG CR 4640, Handbook of Software Quality Assurance Techniques Applicable to the Nuclear Industry
- NUREG/CR- 5147, Fundamental Attributes of a Practical CM Program for Nuclear Plant Design Control
- NUMARC (Nuclear Management & Resources Council)
- NUMARC 90-12, Design Basis Program Guidelines, October 1990

41. Nuclear Science Advisory Committee (NSAC)

- NSAC-105, Guidelines for Design & Procedure Changes in Nuclear Power Plants

42. National Technical Information Service (NTIS)

- ADA076542 CM
- ADA083205 Software CM
- NEI (Nuclear Energy Institute) see: http:www.ac-incorp.com/CM_standards.html

43. NUREG

- NUREG BR 0167, Software Quality Assurance Programs & Guidelines
- NUREG 1000, Generic Implications of ATWS Events at the Salem Nuclear Power Plant
- NUREG/CR 1397, An Assessment of Design Control practices & Design Reconstitution Programs in the Nuclear Power Industry
- NUREG CR 4640, Handbook of Software Quality Assurance Techniques Applicable to the Nuclear Industry
- NUREG/CR- 5147, Fundamental Attributes of a Practical CM Program for Nuclear Plant Design Control
- NUMARC (Nuclear Management & Resources Council)
- NUMARC 90-12, Design Basis Program Guidelines, October 1990

44. Occupational Safety & Health Administration (OSHA) (U.S. Department of Labor)

- OSHA 1910.119, Process Safety Management of Highly Hazardous Chemicals
- OSHA Standards for the Construction Industry (20 CFR Part 1926)
- OSHA Standards for General Industry (29 CFR Part 1910)

45. Radio Technical Commission for Aeronautics (RTCA)

- RTCA/DO-254: Design Assurance Guidance for Airborne Electronic Hardware
- RTCA/d0-178B: Software Considerations in Airborne Systems and Equipment Certification

46. Society of Automotive Engineers (SAE)

- AS9000, Aerospace Basic Quality System Standard (aerospace version of ISO 9000)
- AS9100C: Quality Systems Aerospace - Model for Quality Assurance in Design, Development, Production, Installation, and Servicing

47. Software Engineering Institute (SEI)
http://www.sei.cmu.edu

- Capability Maturity Model Integration (CMMI), Continuous
- Capability Maturity Model Integration (CMMI) Staged

48. Spain

- AENOR UNE-EN 13290-5, Space Project Management—General Requirements—Part 5: Configuration Management
- AENOR UNE 135460-1-1, Road Equipment. Traffic Control Centers. Part 1-1: Remote Stations, Services Management. Communications and Configuration Services
- AENOR UNE 73101, Configuration Management in Nuclear Power Plants

49. Simple Protocol for Independent Computing Environments (SPICE)

- Software Process Improvement and Capability determination, various documents incorporated into ISO/IEC TR 15504:1998

50. TechAmerica

- ANSI/EIA-649B, Configuration Management
- GEIA-HB-649, Configuration Management Handbook
- GEIA-859A, Data Management
- TECHAMERICA CMB 4-1A, Configuration Management Definitions for Digital Computer Programs (withdrawn)
- TECHAMERICA CMB 5-A, Configuration Management Requirements for Subcontractors/ Vendors
- TECHAMERICA CMB 6-10, Education in Configuration and Data Management
- TECHAMERICA CMB 6-1C, Configuration and Data Management References
- TECHAMERICA CMB 6-2, Configuration and Data Management In-House Training Plan
- TECHAMERICA CMB 6-9, Configuration and Data Management Training Course
- TECHAMERICA CMB 7-1, Electronic Interchange of Configuration Management Data
- TECHAMERICA CMB 7-2, Guideline for Transitioning Configuration Management to an Automated Environment
- TECHAMERICA CMB 7-3, CALS Configuration Management SOW and CDRL Guidance
- TECHAMERICA GEIA-TB-0002, System Configuration Management Implementation Template (Oriented for a U.S. Military Contract Environment)

51. U.K Ministry of Defense

- 00-22, The Identification & Marking of Programmable Items
- AVP 38, Configuration Control, Section 3
- MODUK DEF STAN 02-28, Configuration Management Nuclear Submarines In Service Support
- MOD Def Stan 05-57, Configuration Management Policy and Procedures for Defense Material
- NES 41, Requirements for CM & Ship Fit Definitions

FEEDBACK FORM

Title of Document:
Systems Engineering Guide

Technical Report Number:
HA #201701/ISBN

Author
David C. Hall, ESEP/CISSP, Hall Associates LLC

Send To:
Dave.Hall@hallassociateshsval.com

Comment Resolution Matrix

Location of Comment (Para, Section, Page)	**Description of Issue, Comment and Rationale**	**Importance Rating** R- Required, I – Important, T – Think About for Future Version