Chapter 4
Systems Engineering

4.0. Chapter Overview

DoD policy and guidance recognize the importance of and introduce the application of a systems engineering approach in achieving an integrated, balanced system solution. DoD Directive 5000.1 requires:

> ***Systems Engineering**. Acquisition programs shall be managed through the application of a systems engineering approach that optimizes total system performance and minimizes total ownership costs. A modular open-systems approach shall be employed, where feasible.*

DoD Instruction 5000.2 emphasizes the use of systems engineering per the following extract:

> *Effective sustainment of weapon systems begins with the design and development of reliable and maintainable systems through the continuous application of a robust systems engineering methodology.*

Finally, pending inclusion in the next revision to DoD Instruction 5000.2, a memorandum by the USD(AT&L) establishes systems engineering policy and mandates a Systems Engineering Plan for all programs. An extract from the memorandum follows:

> *Systems Engineering (SE). All programs responding to a capabilities or requirements document, regardless of acquisition category, shall apply a robust SE approach that balances total system performance and total ownership costs within the family-of-systems, systems-of-systems context. Programs shall develop a Systems Engineering Plan (SEP) for milestone Decision Authority (MDA) approval in conjunction with each Milestone review, and integrated with the Acquisition Strategy. This plan shall describe the program's overall technical approach, including processes, resources, metrics, and applicable performance incentives. It shall also detail the timing, conduct, and success criteria of technical reviews.*

4.0.1. Purpose

The purpose of this chapter is to facilitate compliance with the above mandatory systems engineering direction. This chapter describes systems engineering processes and the fundamentals of their application to DoD acquisition. It addresses the system design issues that a program manager must face to achieve the desired balanced system solution. In its entirety, this chapter thereby provides guidance and describes expectations for completing the Systems Engineering Plan.

4.0.2. Contents

This Chapter begins with Section 4.1, *Systems Engineering in DoD Acquisition*. This section defines systems engineering and its relationship to acquisition. It also provides perspective on the use of systems engineering processes to translate user-defined capabilities into actionable engineering specifications and on the role of the program manager in integrated system design activities.

Section 4.2, *Systems Engineering Processes: How Systems Engineering is Implemented*, discusses systems engineering processes and activities. The section groups systems engineering processes into technical management processes and technical process categories. This section contains a discussion of the use and tailoring of process models and standards, as well as what to expect of the contractor's systems engineering process.

Section 4.3, *Systems Engineering in the System Life Cycle*, provides an integrated technical framework for systems engineering processes throughout the acquisition phases of a system's life cycle, distinguishing the particular systems engineering inputs and outputs of each acquisition phase.

Section 4.4, *Systems Engineering Decisions: Important Design Considerations*, discusses the many design considerations that should be taken into account throughout the systems engineering processes. This includes an introduction to open systems design; interoperability; software; commercial off-the-shelf items; manufacturing capability; quality; reliability, availability and maintainability; supportability; human systems integration; environment, safety and occupational health; survivability; corrosion prevention and control; disposal and demilitarization; information assurance; insensitive munitions; anti-tamper provisions; system security; and accessibility.

Section 4.5, *Systems Engineering Execution: Key Systems Engineering Tools and Techniques*, includes the important technical, cost, and schedule oversight methods and techniques used in the technical management and technical processes. This section also discusses general knowledge management tools.

Section 4.6, *Systems Engineering Resources*, provides links to many systems engineering resources that already exist across the government, industry, and academia. Links to resources will be incorporated throughout the text of this chapter, as appropriate. As a compilation of available resources, this section includes standards and models, handbooks and guides, as well as any additional references deemed appropriate.

4.1. Systems Engineering in DoD Acquisition

Systems engineering is the overarching process that a program team applies to transition from a stated capability need to an operationally effective and suitable system. Systems engineering encompasses the application of systems engineering processes across the acquisition life cycle (adapted to each and every phase) and is intended to be the integrating mechanism for balanced solutions addressing capability needs, design considerations and constraints, as well as limitations imposed by technology, budget, and schedule. The systems engineering process is applied early in concept definition, and then continuously throughout the total life cycle.

Balanced system solutions are best achieved by applying established systems engineering processes to the planning, development, and implementation of a system or system-of-systems acquisition in an Integrated Product and Process Development framework.

4.1.1. Systems Engineering

Systems engineering is an interdisciplinary approach or a structured, disciplined, and documented technical effort to simultaneously design and develop systems products and processes to satisfy the needs of the customer. Systems engineering transforms needed operational capabilities into an integrated system design through concurrent consideration of *all* life-cycle needs. As systems become larger and more complex, the design, development, and production of a system or system-of-systems require the integration of numerous activities and processes. Systems engineering is the approach to coordinate and integrate all acquisition life-cycle activities. Systems engineering integrates diverse technical management processes to achieve an integrated systems design. Although numerous definitions exist, this chapter adopts the following formal definition, adapted from EIA/IS 632, *Processes for Engineering a System*:

> *Systems engineering is an interdisciplinary approach encompassing the entire technical effort to evolve and verify an integrated and total life-cycle balanced set of system, people, and process solutions that satisfy customer needs. Systems engineering is the integrating mechanism across the technical efforts related to the development, manufacturing, verification, deployment, operations, support, disposal of, and user training for systems and their life cycle processes. System engineering develops technical information to support the program management decision-making process. For example, systems engineers manage and control the definition and management of the system configuration and the translation of the system definition into work breakdown structures.*

Systems engineering provides a systematic set of processes to help coordinate and integrate activities throughout the life cycle of the system. Systems engineering offers a technical framework to enable sound decision making relative to trade studies among system performance, risk, cost, and schedule. The successful implementation of proven, disciplined systems engineering processes results in a total system solution that is—

- Robust to changing technical, production, and operating environments;
- Adaptive to the needs of the user; and
- Balanced among the multiple requirements, design considerations, design constraints, and program budgets.

Systems engineering is a broad topic. Before this Guidebook goes into the full technical detail of implementing systems engineering, we will introduce the various participant's responsibilities in systems engineering, discuss the "total systems approach" and "total life cycle systems management" required by DoD Directive 5000.1, relate systems engineering to the IPPD process, and recommended systems engineering leadership practices.

4.1.2. Participants in Systems Engineering

The program manager should implement a robust systems engineering approach to translate operational needs and capabilities into operationally suitable increments of a system. Systems engineering permeates design, production, test and evaluation, and system support. Systems engineering principles should influence the balance among the performance, cost, and schedule parameters and associated risks of the system. Program managers exercise leadership, decision-making, and oversight throughout the system life cycle. Implementing a systems engineering approach adds discipline to the process and provides the program manager with the information necessary to make valid trade-off decisions throughout a program's life cycle.

Systems engineering is typically implemented through multi-disciplined teams of subject matter experts (often formally chartered as an Integrated Product Team (IPT). The systems engineering working-level IPT translates user-defined capabilities into operational system specifications consistent with cost, schedule, and performance constraints. (See the DoD Directive 5000.1 discussion of Knowledge Based Acquisition and additional information in this Guidebook.) While the program office usually has a Chief Engineer or Lead Systems Engineer in charge of the systems engineering process, personnel from non-systems engineering organizations or from outside the program management structure may also perform activities related to systems engineering. Most program personnel should see themselves as participants in the systems engineering processes. Systems engineering-like activities include defining architectures and capabilities and conducting functional analyses per CJCS Instruction 3170.01. Warfighters, sponsors, and planners usually complete these activities before a program is initiated.

4.1.3. Total Life Cycle Systems Management (TLCSM) in Systems Engineering

It is fundamental to systems engineering to take a total life cycle, total systems approach to system planning, development, and implementation. Total life cycle systems management (TLCSM) is the planning for and management of the entire acquisition life cycle of a DoD system. Related to the *total systems approach*, DoD Directive 5000.1, E1.29, makes the program manager accountable for TLCSM:

> *E1.29. Total Systems Approach. The PM shall be the single point of accountability for accomplishing program objectives for total life-cycle systems management, including sustainment. The PM shall apply human systems integration to optimize total system performance (hardware, software, and human), operational effectiveness, and suitability, survivability, safety, and affordability. PMs shall consider supportability, life cycle costs, performance, and schedule comparable in making program decisions. Planning for Operation and Support and the estimation of total ownership costs shall begin as early as possible. Supportability, a key component of performance, shall be considered throughout the system life cycle.*

Because of TLCSM, the program manager should consider nearly all systems development decisions in context of the effect that decision will have on the long term operational effectiveness and logistics affordability of the system. TLCSM considerations should permeate the decision making of all acquisition functions and communities, during all acquisition phases. In fact, TLCSM factors should be considered by the participants in the Joint Capabilities Integration and Development System (JCIDS) even before a program manager is assigned; the JCIDS determination of performance capabilities should reflect TLCSM considerations. Later, TLCSM should frame the decision making for sustainment logistics.

TLCSM encompasses the following concepts:

- Single point of accountability;
- Evolutionary acquisition;
- Supportability and sustainment as key elements of performance;
- Performance-based strategies, including logistics;
- Increased reliability and reduced logistics footprint; and

- Continuing reviews of sustainment strategies.

In executing TLCSM responsibilities, program managers should apply systems engineering processes and practices known to reduce cost, schedule, and performance risks. This includes best public sector and commercial practices and technology solutions (see section 4.5.9.1 for links to best practice examples). The resulting system solution should be interoperable and should meet JCIDS and JCIDS-related (e.g., Condition Based Maintenance Plus or affordability) performance capabilities needs. The TLCSM business approach means that all major materiel alternative considerations and major acquisition functional decisions reflect an understanding of the effects and consequences of these decisions on Operations and Sustainment Phase (including disposal) system effectiveness and affordability.

The cost to implement a system change increases as a program moves further along the system life cycle. The greatest leverage exists in the early stages of development, when the program is most flexible. Early in the life cycle, thorough analyses of life-cycle issues and cost/performance trade-off studies can reveal a balanced, life-cycle design that prevents costly changes later in the system life cycle.

The program manager should apply a robust systems engineering methodology to achieve the optimal balance of performance and total ownership costs. Effective sustainment of weapons systems begins with the development of a balanced system solution. The key is to apply the systems engineering processes throughout the DoD 5000 Defense Acquisition Management Framework. Systems engineering should play a principal role in each acquisition phase. See Section 4.3 for a detailed description of these systems engineering activities by acquisition phase.

Consequently, systems engineering should applied at the initial stages of program formulation to provide the integrated technical basis for program strategies; acquisition plans; acquisition decisions; management of requirements, risk, and design trades; and integration of engineering, logistics, test, and cost estimation efforts among all stakeholders. Likewise, the Systems Engineering Plan (SEP) should be established early in the program definition stages and updated periodically as the program matures. The overall systems engineering strategy should be addressed in and integrated with all other program strategies. Systems engineering enables TLCSM, and provides the framework to aid decision making about trade-offs between system performance, cost, and schedule.

4.1.4. Systems Engineering and the New Acquisition Environment

Evolutionary acquisition strategies integrate advanced, mature technologies into producible systems that can be deployed to the user as quickly as possible. An evolutionary acquisition strategy matches available technology and resources to approved, time-phased, capability needs. Systems engineering processes provide the disciplined, integrated development and production environment that supplies increasing capability to a materiel solution. In spiral and incremental development, capability is developed and fielded in increments with each successive increment building upon earlier increments to achieve an overall capability. These approaches to evolutionary acquisition are particularly effective in quickly fielding an initial capability or increment of functionality while allowing continued efforts to incrementally attain the final, full, end-state capability. Robust systems engineering processes ensure that systems are designed to easily and affordably accommodate additive capabilities in subsequent increments. Examples of these processes include the modular, open systems approach and robust design.

There are various development and life-cycle models to support systems engineering within an evolutionary acquisition strategy. They include the waterfall, spiral, and "Vee" models. All models provide an orderly approach to implementing and integrating the systems engineering processes during each acquisition phase. The spiral and Vee models rely heavily on prototyping, both physical and virtual, to get user feedback.

Evolutionary acquisition has increased the importance of traceability in program management. If a defense system has multiple increments, systems engineering can trace the evolution of the system. It can provide discipline to and documentation of the repeated trade-off analyses and decisions associate with the program. Due to the nature of evolutionary acquisition, design, development, deployment, and sustainment can each be occurring simultaneously for different system increments.

4.1.5. The Integrated Product and Process Development (IPPD) Framework and Systems Engineering

The Department of Defense defines IPPD as a management technique that uses multidisciplinary teams (Integrated Product Teams (IPTs)) to optimize design, manufacturing, and supportability processes. IPPD facilitates meeting cost and performance objectives from system concept out through production and field support. It is a broad, interdisciplinary approach that includes not only the engineers, technical specialists, and customers in the IPTs, but also business and financial analysts as well. (See also 10.3, 11.8, and the IPPD Handbook.)

Systems engineering is consistent with IPPD. It creates and verifies an integrated and life-cycle-balanced set of system product and process solutions that satisfy stated customer needs. Systems engineering integrates the development of the system with the development of all system-related processes. The systems engineering process provides a common basis for and improves the communication between IPT members. All members of the development IPTs, who possess expertise in one or more disciplines in a system's life cycle, perform systems engineering; everyone involved in the system's development should be a "total systems-thinker." Each member of the team should apply the systems engineering process to their respective area of expertise.

4.1.6. Systems Engineering Leadership

As part of their overall role in technical oversight of assigned programs, acquisition components should maintain a systems engineering technical authority. A technical authority is the organization outside the program manager's chain of command with responsibility and accountability to establish, approve, and judge conformance of products and technical processes to technical requirements and policy during all phases of product development, acquisition, and sustainment. This technical authority should ensure proper systems engineering process application to programs and ensure proper training, qualification, and oversight of systems engineering personnel assigned to programs. As part of this overall responsibility for technical oversight, the technical authority should:

- Nominate a lead/chief systems engineer to the program manager at the initial stages of program formulation. The lead/chief systems engineer should be accountable to the program manager for meeting program objectives and accountable to the systems engineering technical authority for the proper application of systems engineering, and

- Nominate a chair for program technical reviews that is independent of the assigned program team and approved by the program manager. Technical reviews should include participation by program team personnel and independent (of the program team) subject matter experts as identified by the chair.

4.2. Systems Engineering Processes: How Systems Engineering is Implemented

This section discusses the use and tailoring of process models and standards, presents the program office systems engineering processes as management processes and technical processes, and describes common expectations of the contractor's systems engineering process.

4.2.1. Processes Overview

Overall, the flow of the systems engineering processes is iterative within any one phase of the acquisition process and is recursive at lower and lower levels of the system structure. Systems engineering processes are applied to allow an orderly progression from one level of development to the next more detailed level through the use of controlled baselines. These processes are used for the system, subsystems, and system components as well as for the supporting or enabling systems used for the production, operation, training, support, and disposal of that system. During the course of technical management processes and activities, such as trade studies or risk management activities, specific requirements, interfaces, or design solutions may be identified as non-optimal and changed to increase system-wide performance, achieve cost savings, or meet scheduling deadlines. The value of these processes is not only the transition of requirements from design to system, but as an integrated framework within which the universe of requirements can be, as a collective whole, defined, analyzed, decomposed, traded, managed, allocated, designed, integrated, tested, fielded, and sustained.

4.2.2. Standards and Models

Many systems engineering process standards and models exist that describe best practice in accomplishing systems engineering. The program manager should use these process standards and models as appropriate on the program. These models usually contain guidance for tailoring, which is best done in conjunction with a risk assessment on the program that leads the program manager to determine which specific processes and activities are vital to the program. Some examples of systems engineering process standards and models include the following:

- ISO/IEC 15288, *Systems Engineering—System Life Cycle Processes*
- ANSI/EIA 632, *Processes for Engineering a System*
- IEEE 1220, *Application and Management of the Systems Engineering Process*
- EIA 731, *Systems Engineering Capability Model*
- CMMI SWE/SE/IPPD/SS, *Capability Maturity Model-Integration for Software Engineering, Systems Engineering, Integrated Product and Process Development and Supplier Sourcing*

4.2.2.1. Primary Standards

Three primary systems engineering standards represent different levels of application:

- The International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15288, *Systems Engineering—System Life Cycle Processes*, covers the life cycle of a man-made system from concept through retirement. "It provides the processes for acquiring and supplying system products and services that are configured from one or more of the following types of system components: hardware, software, and humans. In addition, the framework provides for the assessment and improvement of the life cycle."[1] This standard is designed to be used by an organization, a project within an organization, or an acquirer and a supplier via an agreement.
- The Electronic Industry Alliance (EIA) 632, *Processes for Engineering a System*, defines the set of requirements for engineering a system. The processes in EIA 632 describe "what to do" with respect to the processes for engineering a system, which is the next level down from the ISO/IEC 15288 level of system life cycle processes.
- The Institute of Electrical and Electronic Engineers (IEEE) 1220 defines a systems engineering process. It gives the next level of detail below the process requirements described in EIA 632. The process is described more at the task or application level. IEEE 1220 does not worry about "who does what" as some of the other standards do with the "acquirer-supplier" concepts.

To actually accomplish systems engineering, an organization would most likely need all three standards or a hybrid model of their own.

4.2.2.2. Standardized Terminology

The many systems and software engineering process models and standards use different terms to describe the processes, activities, and tasks within the systems engineering and other life-cycle processes. This chapter uses the following terminology to represent generic systems engineering processes. They are grouped in two categories: Technical Management Processes and Technical Processes:

- Technical Management Processes

[1] ISO/IEC 15288, Introduction.

- Decision Analysis
- Technical Planning
- Technical Assessment
- Requirements Management
- Risk Management
- Configuration Management
- Technical Data Management
- Interface Management

- Technical Processes
  - Requirements Development
  - Logical Design Solution
  - Physical Design Solution
  - Implementation
  - Integration
  - Verification
  - Validation
  - Transition

These generic processes are described briefly below and applied throughout the life-cycle phases. More detail with regard to systems engineering processes can be found in any of the above-mentioned standards or models. Since systems engineering cannot be conducted without good organization and project processes as well as sufficient infrastructure, these standards and models also may include processes and activities, such as organizational training, that are beyond the technical ones that may be considered specific to systems engineering.

4.2.3. Technical Management Processes

The program manager uses technical management processes to *manage* the technical development of the system increments, including the supporting or enabling systems. Section 4.5 describes the key techniques and tools for technical management in detail.

4.2.3.1. Decision Analysis

Decision Analysis activities provide the basis for evaluating and selecting alternatives when decisions need to be made. Decision Analysis involves selecting the criteria for the decision and the methods to be used in conducting the analysis. For example, during the Design Solution process, decisions among alternative designs should achieve a balanced, operationally effective and suitable, robust, system solution. The decision analysis supporting these decisions could include trade studies, the implementation of risk mitigation decisions, the use of robust design techniques, the use of models and simulations to perform analyses, and augmenting the analysis with virtual and physical prototypes. Decision criteria should be influenced by interoperability constraints, open system factors, affordability, schedule, etc. As another example, iteration with the Requirements Development and Implementation processes is necessary to decide whether to make a new system element, buy an existing one, or reuse one you already have. Cost and schedule considerations, the evolutionary acquisition strategy, and interface requirements, among other factors, would influence the criteria for this kind of decision.

4.2.3.2. Technical Planning

Technical Planning activities ensure that the systems engineering processes are applied properly throughout a system's life cycle. Technical planning, as opposed to program planning, addresses the scope of the technical effort required to develop the system. A mandated tool for this activity is the Systems Engineering Plan. Each of the technical processes requires technical planning. Technical planning for Implementation, Integration, Verification, Validation, and

Transition processes, and their accompanying systems, can reveal constraints and interfaces that will result in derived technical requirements.

4.2.3.3. Technical Assessment

Technical Assessment activities measure technical progress and the effectiveness of plans and requirements. Activities within Technical Assessment include the activities associated with Technical Performance Measurement and the conduct of technical reviews. A structured review process should demonstrate and confirm completion of required accomplishments and exit criteria as defined in program and system planning. Technical reviews are discussed in detail in section 4.3. Technical assessment activities discover deficiencies or anomalies that often result in the application of corrective action.

4.2.3.4. Requirements Management

Requirements Management provides traceability back to user-defined capabilities as documented through the Joint Capabilities Integration and Development System. In evolutionary acquisition, the management of requirements definition and changes to requirements takes on an added dimension of complexity. The program manager should institute Requirements Management to (1) maintain the traceability of all requirements from capabilities needs, (2) to document all changes to those requirements, and (3) to record the rationale for those changes. Emerging technologies and threats can influence the requirements in the current as well as future increments of the system.

4.2.3.5. Risk Management

Risk management in systems engineering examines the risks of deviating from the program plan. It examines all aspects of the program, from conception to disposal, early in the program and in relation to each other. Most risk management approaches have in common the practice of integrating design (performance) requirements with other life-cycle issues such as manufacturing, operations, and support.

The program manager establishes a risk management process, including planning, assessment (identification and analysis), handling, and monitoring, to be integrated and continuously applied throughout the program, including, but not limited to, the design process. The risk management effort addresses:

- Risk planning;
- Risk assessment;
- Risk handling and mitigation strategies; and
- Risk monitoring approaches.

Risk assessment includes identification and analysis of potential sources of risk to the program plan, including, but not limited to, cost, performance, and schedule risks based on such factors as:

- The technology being used and its related design;
- Manufacturing capabilities;
- Potential industry sources; and

- Test and support processes.

The overall risk management effort interfaces with technology transition planning, including the establishment of transition criteria for such technologies.

More specifically, technology transfer risk management is a systematic methodology to identify, evaluate, rank, and control inadvertent technology transfer. It is based on a three-dimensional model: the *probability* of occurrence, the *consequence* if realized, and *countermeasure cost* to mitigate the occurrence. This is a key element of a program manager's executive decision-making – maintaining awareness of technology alternatives and their potential sensitivity while making trade-off assessments to translate desired capabilities into actionable engineering specifications. To successfully manage the risk of technology transfer, the program manager should:

- Identify contract vehicles which involve the transfer of sensitive data and technology to partner suppliers;
- Evaluate the risks that unfavorable export of certain technologies could pose for the program; and
- Develop alternatives to mitigate those risks (see also section 8.4).

More information can be found in the *DoD Risk Management Guide*.

4.2.3.6. Configuration Management

Configuration Management (CM) (See DoD Directive 5000.1) is the application of sound business practices to establish and maintain consistency of a product's attributes with its requirements and product configuration information. It involves interaction among government and contractor program functions such as systems engineering, design engineering, logistics, contracting, and manufacturing in an Integrated Product Team environment. A configuration management process guides the system products, processes, and related documentation, and facilitates the development of open systems. CM efforts result in a complete audit trail of decisions and design modifications. The elements of configuration management include:

- Configuration Management Planning and Management -- Provides total life cycle configuration management planning for the program/project and manages the implementation of that planning;
- Configuration Identification -- Establishes a structure for products and product configuration; selects, defines, documents, and baselines product attributes; and assigns unique identifiers to each product and product configuration information item;
- Configuration Change Control -- Ensures that changes to a configuration baseline are properly identified, recorded, evaluated, approved or disapproved, and incorporated and verified, as appropriate;
- Configuration Status Accounting -- Manages the capture and maintenance of product configuration information necessary to account for the configuration of a product throughout the product life cycle; and
- Configuration Verification and Audit -- Establishes that the performance and functional requirements defined in the product definition information have been achieved by the

design and that the design has been accurately documented in the product definition information.

Some examples of configuration management process standards and best practices are:

- ANSI/EIA 649A, Configuration Management, on the GEIA website (Click on STANDARDS);
- ISO 10007, Quality Management – Guidelines for Configuration Management; and
- EIA 836, Configuration Management Data Exchange and Interoperability, located on the GEIA website (Click on STANDARDS).

4.2.3.7. Data Management

Data are defined as recorded information regardless of the form or method of recording. The term includes technical data, computer software documentation, management information, representation of facts, numbers, or datum of any nature that can be communicated, stored, and processed to form information required by a contract or agreement to be delivered, or accessed by, the Government. The term includes similar information generated directly by Government activities, as well. The data are used to gain insight and provide management and guidance to systems development programs.

For purposes of this chapter, "data" refers to the information necessary for or associated with product development and sustainment, including the data associated with system development; modeling and simulation used in development or test; test and evaluation; installation; parts; spares; repairs; usage data required for product sustainment; and source and/or supplier data. Data specifically not included would be data relating to tactical operations information; sensor or communications information; financial transactions; personnel data; transactional data; and other data of a purely business nature. Guidance for logistics data can be found in section 5.1.3.3.

Data Management plays an important role in the systems engineering process. In the program office, data management consists of the disciplined processes and systems used to plan for, acquire, access, manage, protect, and use data of a technical nature to support the total life cycle of the system. Under the Total Life Cycle Systems Management concept, the program manager is responsible for Data Management. The program manager should develop a plan for managing defense system data during each phase of the system life cycle and include it in the Systems Engineering Plan.

Data Management applies policies, systems, and procedures to identify and control data requirements; to responsively and economically acquire, access, and distribute data; and to analyze data use. Adherence to data management principles enables the sharing, integration, and management of data by government and industry, and ensures that data products (information) meet or exceed customer requirements. Recent government and industry initiatives in Data Management have changed the approach and scope of data management, and made it a stronger element in the systems engineering process.

Data Management has a leading role in capturing, organizing, and providing information for the following uses in the systems engineering process:

- Enabling collaboration and life cycle use of acquisition system product data;
- Capturing and organizing all systems engineering inputs, as well as current, intermediate, and final outputs;
- Providing data correlation and traceability among requirements, designs, solutions, decision, and rationale;
- Documenting engineering decisions, including procedures, methods, results, and analyses;
- Functioning as a reference and support tool for the systems engineering effort and process;
- Facilitating technology insertion for affordability improvements during re-procurement and post-production support; and
- Supporting configuration procedures, as needed.

Examples of Data Management process standards and guidance documents are listed below:

- S1000D International Specification for Technical Publications Utilizing a Common Source Database;
- Data Management Community of Practice (CoP), located on the Acquisition Community Connection on the DAU website;
- DoD 5010.12-M, Procedures for the Acquisition and Management of Technical Data, May 1993;
- DoD 5200.1-M Acquisition System Protection Program, March 1994;
- GEIA-859, Consensus Standard for Data Management, located on the GEIA website (Click on STANDARDS). (Note: This document is currently being published.);
- Intellectual Property: Navigating Through Commercial Waters, October 15, 2001, website;
- ISO 10303, Standard for the Exchange of Product Model Data (STEP).

The program manager should develop a plan for managing defense system data during each phase of the system life cycle. Government inspection and acceptance is required for technical publications, product definition data elements, and other data that will be used by DoD Component personnel for the installation, operation, or maintenance of equipment or software. Establishing data exchange formats promotes data reuse, fosters competition, and helps to ensure that data can be used consistently throughout the system, family of systems, or system of systems.

4.2.3.7.1. Data Acquisition

Defense system data are acquired when needed to support the acquisition, operations, maintenance, or disposal of the system. The applied systems engineering process requires *access* to data to facilitate decision making, but does not necessarily require *acquisition* of *all* data. The data management processes assist in decision-making. Data management processes reveal the proper data to be acquired or accessed. The decision to purchase data should be made when access to required data is not sufficient to provide for life-cycle planning and system maintenance. The cost of data delivery should be a primary consideration. Other considerations include the following:

- Data requirements for spare and repair parts;
- Technical data needed for ordering and purchasing items for contingencies; and
- Circumstances under which the data may evolve over time to more useful or updated data.

4.2.3.7.2. Data Protection

The program manager is responsible for protecting system data, whether the data are stored and managed by the government or by contractors. The DoD policy with regard to data protection, marking, and release can be found in DoD Directive 5230.24, DoD Directive 5230.25, and DoD 5400.7-R. Data containing information subject to restrictions are required to be protected in accordance with the appropriate guidance, contract, or agreement. Guidance on restriction statements can be found in the DFARS Part 252.227-7013 & 7014, and DoD Directive 5230.24. When digital data are used, the data should display applicable restriction markings, legends, and distribution statements clearly visible when the data is first opened or accessed. These safeguards not only assure government compliance with use of data, they also guarantee and safeguard contractor data that are delivered to the government, and extend responsibilities of data handling and use to parties who subsequently use the data.

All data deliverables should include distribution statements and processes should be established to protect all data which contain critical technology information, as well as assure that limited distribution data, intellectual property data, or proprietary data are properly handled during systems engineering activities – whether the data are hard copy or digital.

4.2.3.7.3. Data Storage

The program manager also has responsibility for addressing long-term storage and retrieval of data and associated program information – planning for digitizing continued need information, as appropriate and cost-effective. Such long-term planning and incremental digitization, as required, will assure that applicable data is available, preserved, and migrated to successive formats for future planning and use.

4.2.3.8. Interface Management

The Interface Management process ensures interface definition and compliance among the elements that compose the system; as well as with other systems with which the system or system elements must interoperate. Interface management control measures ensure that all internal and external interface requirement changes are properly documented in accordance with the configuration management plan and communicated to all affected configuration items.

Many of the external interfaces are identified through the JCIDS process and its accompanying documents and architectures. As system interface control requirements are developed, they are documented and made available to the appropriate Integrated Product Team. Documented interface control requirements serve critical functions at all levels of the system. Some of these functions include the following: to facilitate competitive bids; to enable integration of system and sub-systems; to support system maintenance, future enhancement, and upgrades; and provide input data for continuous risk management efforts. Refinement of the interfaces is achieved through iteration. As more is learned about the system during the design phases, lower-level, verifiable requirements and interfaces are defined and refined. Impacts to the original defined capabilities and interfaces, performance parameter thresholds and objectives, and the system are evaluated when defining and modifying interfaces.

4.2.4. Technical Processes

The program manger uses technical processes to design the system, subsystems, and components, including the supporting or enabling systems required to produce, support, operate, or dispose of a system. (The terminology used to indicate a subsystem is system element, component, or configuration item, depending on the systems engineering context and phase of acquisition under discussion.) Section 4.5 discusses some key techniques and tools for conducting the analyses required in technical processes.

4.2.4.1. Requirements Development

The Requirements Development process takes all inputs from relevant stakeholders and translates the inputs into technical requirements. DoD systems engineers primarily respond to the Joint Capabilities Integration and Development System (JCIDS) documents that identify capability gaps in need of a materiel solution. The program manager should work with the user to establish and refine operational needs, attributes, performance parameters, and constraints that flow from JCIDS-described capabilities, and then ensure that all relevant requirements are addressed (see Figure 21, System Operational Effectiveness Diagram of Section 4.4). Together with the user, the program manager should translate "customer needs" into the following program and system requirements:

- Performance parameter objectives and thresholds;
- Affordability constraints;
- Scheduling constraints; and
- Technical constraints.

Since some of the requirements may become defined only through system decomposition at later stages of the program, iterative application of rigorous systems engineering is key.

Requirements Development encompasses the definition and refinement of system-, subsystem-, and lower-level functional and performance requirements and interfaces to facilitate the design of open systems. It allocates and balances interoperability requirements among systems that should interoperate successfully to satisfy all appropriate integrated architectures and CRDs[2] under which the proposed system falls.

An integral part of defining and refining requirements is to provide technical support to the market research required early in the program life cycle. Systems engineers within DoD face the same sorts of requirements definition tasks that their commercial counterparts encounter in

---

[2] Although integrated architectures will replace the Capstone Requirements Documents for systems of systems, the Capstone Requirements Document will be used until the architectures are in place.

addressing market research (and customer needs). These tasks involve analyzing if and how an existing product (commercial or non-developmental item) can meet user requirements. This analysis ensures that open systems principles are applied to the maximum extent possible to reduce both life-cycle costs and development cycle time.

Requirements Development complements Logical Solution and Design Solution technical processes. These three processes are iterated at each level of the system structure, and then applied recursively to lower levels of the physical architecture throughout development. The objective is to help ensure that the requirements derived from the customer-designated capabilities are feasible and effective, as well as updated, as more information is learned about the requirements and interfaces through analysis.

4.2.4.2. Logical Design Solution

Logical Design Solution is the process of obtaining sets of logical solutions to improve understanding of the defined requirements and the relationships among the requirements (e.g., functional, behavioral, temporal). Once the logical solution sets are formed, the engineers allocate performance parameters and constraints, and then define derived technical requirements to be used for the system design.

There are many ways to attain the logical solution sets. Traditionally, the Department of Defense has used functional analysis/allocation. However, other approaches, such as behavioral analysis, timeline analysis, object-oriented analysis, data-flow analysis, and structured analysis, may also apply.

The design approach resulting from logical solution:

- Partitions a system into self-contained, cohesive, logical groupings of interchangeable and adaptable elements to enable ease of change, achieve technology transparency and mitigate the risk of obsolescence
- Uses rigorous and disciplined definitions of interfaces and, where appropriate, defines the key interfaces within a system by widely supported standards (including interface standards, protocols, and data interchange language and standards) that are published and maintained by recognized standards organizations

When using a functional approach, the output of this process is the functional architecture that puts all of the functions in order, thereby sequencing all of the system tasks that should occur. The functional architecture provides a functional “picture” of the system. It details the complete set of functions to be performed along with the relationships among the functions.

4.2.4.3. Physical Design Solution

The Physical Design Solution process translates the outputs of the Requirements Development and Logical Design Solution processes into alternative design solutions and selects a final design solution. The alternative design solutions include—

- People, products, and process entities and
- Related internal and external interfaces.

Not only does this process iterate with Requirements Development and Logical Design Solution, it also integrates with the program decision processes to identify and select the best solution. If the process finds that specified objectives and thresholds are infeasible, ineffective, or result in an inefficient system, it may then be necessary to re-evaluate the defined performance parameters.

The output of this process is the design or physical architecture that forms the basis for design definition documentation such as specifications, baselines, and Work Breakdown Structures (WBS). Physical architectures should be sufficiently detailed to allow the following:

- Confirmation of upward and downward traceability of requirements;
- Confirmation of interoperability and open system performance requirements; and
- Demonstration of the appropriate products to satisfy the applicable acquisition phase exit criteria.

Confirmation of requirements traceability and the soundness of the selected physical architecture can be accomplished using a cost-effective combination of design analysis, design modeling, and simulation, as applicable.

4.2.4.4. Implementation

Implementation is the process that actually yields the lowest level system elements in the system hierarchy. The system element is made, bought, or reused. Making it involves the hardware fabrication processes of forming, removing, joining, and finishing; or the software processes of coding, etc. If implementation involves a production process, a manufacturing system is required to be developed using these same technical and technical management processes.

Depending on the technologies and systems chosen when a decision is made to produce a system element, the Implementation process imposes constraints on the Design Solution process. If the decision is made to purchase or reuse an existing system element, the Implementation process may involve some adaptation or adjustments to the system element. The Implementation process gets the system element ready for the processes of Integration, Verification, and Validation. It should include some testing of the implemented system element before the element passes to the Integration Process. Implementation may also involve packaging, handling, and storage, depending on where or when the system element needs to be integrated into a higher-level assembly. Developing the supporting documentation for the system element—such as the manuals for operations, maintenance, and/or installation—are also a part of the Implementation process.

4.2.4.5. Integration

Integration is the process of incorporating the lower-level system elements into a higher-level system element in the physical architecture. The plan or strategy for the Integration process, including the assembly sequence, may impose constraints on the design solution. An assembled system element, also developed with the technical and technical management processes, may include fixtures for hardware or compilers for software.

Integration also refers to the incorporation of the final system into its operational environment and defined external interfaces.

Interface Management plays an important role with Integration, and iteration between the two processes will occur.

4.2.4.6. Verification

The Verification process confirms that the system element meets the design-to or build-to specifications. It answers the question "Did you build it right?" As such, it tests the system elements against their defined requirements ("build-to" specifications). The purpose of Verification is to:

- Conduct verification of the realized (implemented or integrated) system element (including interfaces) from the lowest level system element up to the total system to ensure that the realized product conforms to the build-to specifications;
- Generate evidence necessary to confirm that system elements at each level of the system hierarchy meet their build-to specifications; and
- Verify the materials employed in system solutions can be used in a safe and environmentally compliant manner.

The nature of verification activities changes as designs progress from concept to detailed designs to physical products. Throughout the system's life cycle, however, design solutions at all levels of the physical architecture are verified through a cost-effective combination of analysis, examination, demonstration, and testing, all of which can be aided by modeling and simulation.

4.2.4.7. Validation

The Validation process answers the question of "Did you build the right thing?" As such, it tests the performance of systems within their intended operational environment, with anticipated operators and users. In the early stages of the system life cycle, validation may involve prototypes, simulations, or mock-ups of the system and a model or simulation of the system's intended operational environment.

4.2.4.8. Transition

Transition is the process applied to move the system element to the next level in the physical architecture or, for the end-item system, to the user. This process may include installation at the operator or user site.

4.2.5. The Contractor's Systems Engineering Process

Contractor selection should depend on demonstrated process capability and organizational maturity in their systems engineering processes, as well as on demonstrated domain expertise and past performance commensurate with the needs of the program. Organizations use different standards and models and their accompanying assessment methods to establish the initial capability of their systems engineering processes and then to improve those processes. Some of the different standards and models for systems engineering were discussed in section 4.2.2. The remainder of this section covers some of the things a program manager needs to know when a contractor uses these systems engineering standards or models and their accompanying methods for appraisals and assessments.

4.2.5.1. The Use of Standards versus Capability and Maturity Models

The major distinction between standards and capability and maturity models lies in their purpose. Standards provide recommended processes to apply within an organization, describe expected tasks and outcomes, and describe how the processes and tasks integrate to provide required inputs and outputs. Standards are meant to provide an organization with a set of processes that, if done by qualified persons using appropriate tools and methods, will provide a capability to do effective and efficient engineering of systems. Capability and maturity models, on the other hand, are for process improvement. Capability and maturity models are used to assess, from an organizational perspective, how well the standard processes are being performed. Both capability and maturity models and standard processes are useful to an organization, but the role for each should be kept in perspective. The solicitation effort should seek descriptions of potential offerors' models and standards.

In general, the program manager should ensure that the contractor has established a process or processes to conduct systems engineering, that the contractor maintains these processes, and that throughout the organization, work adheres to these processes. Selecting an offeror with a weak systems engineering process will likely result in problems such as poor understanding of requirements and design constraints and how these are managed, little or no system design evolution documentation, poor configuration control, and inadequate manufacturing quality control.

4.2.5.2. Capability Reviews

Capability reviews such as manufacturing capability and software capability reviews are a useful tool available during source selections to assess the offerors' capability in selected critical process areas. Capability reviews may be the appropriate means for evaluating program-specific critical processes such as systems engineering, software development, configuration management, etc. The reviews would be useful to supplement process past performance data to ascertain the risks in selecting a given offeror and to assist in establishing the level of government oversight needed to manage the process-associated risks if that offeror is awarded the contract. The trade-off in determining whether or not to do a capability review would be the criticality of the process versus the time and resources to do the review versus the availability, adequacy, and currency of an offeror's process past performance data.

4.2.5.3. Capability Appraisals

In all cases, the program manager retains the right (and is encouraged) to independently evaluate the process capabilities of the selected team prior to or immediately after contract award in order to have a better understanding of potential risks associated with the development team's process capabilities. Once the developer is selected, the program manager can conduct an evaluation to support the up-front risk assessment of the developer's capability to deliver.

Periodic appraisals are encouraged as part of contract process monitoring activities. The selection of assessment or appraisal method would be dependent upon the needs of the particular project, the level of risk associated with the project, and any areas of concern the program manager may have. The program manager should understand that: 1) appraisal and assessment results are another tool (like past performance) to gauge the likelihood that the contractor will succeed and perform to the requirements of the contract; 2) assessments are most valuable when they apply across the full program team, and not just one segment of the organization; and 3) domain experience is at least as important as process maturity level when evaluating the program team's capability.

4.2.6. System of Systems Engineering

System of systems engineering deals with planning, analyzing, organizing, and integrating the capabilities of a mix of existing and new systems into a system of systems capability greater than the sum of the capabilities of the constituent parts. It is a top-down, comprehensive, collaborative, multidisciplinary, iterative, and concurrent technical management process for identifying system of systems capabilities; allocating such capabilities to a set of interdependent systems; and coordinating and integrating all the necessary development, production, sustainment, and other activities throughout the life cycle of a system of systems. The overall objective for developing a system of systems is to satisfy capabilities that can only be met with a mix of multiple, autonomous, and interacting systems. The mix of constituent systems may include existing, partially developed, and yet-to-be-designed independent systems. Systems of systems should be treated and managed as a system in their own right, and should therefore be subject to the same systems engineering processes and best practices as applied to individual systems.

The engineering of a system of systems differs from the engineering of a single system. The set of systems comprising the system of systems are independently useful systems, yet when integrated together, they deliver significantly improved capability. A single system or less than full combination of all systems cannot provide the capability achieved by the system of systems.

The consideration of system of systems engineering should include the following factors or attributes:

- Larger scope and greater complexity of integration efforts;
- Collaborative and dynamic engineering;
- Engineering under the condition of uncertainty;
- Emphasis on design optimization;
- Continuing architectural reconfiguration;
- Simultaneous modeling and simulation of emergent system of systems behavior; and
- Rigorous interface design and management.

<u>System of Systems Engineering Implications for Single System Developers</u>. Systems should not be developed as stand-alone systems, but as parts of larger meta-systems delivering unique and encompassing capabilities. Program managers should be aware of the distinguishing system of systems engineering attributes that might apply to their system and the possible impact on their system architecture. Program managers should use the following list of questions to address system of systems concerns, capitalize on system of systems capability pay-offs, and

effectively meet the design and development requirements of current and future system of systems:

1. Will joint warfighting capabilities improve if the Department incorporates my system into the portfolio of existing and planned systems of systems?

2. What additional capabilities and behavior could my system deliver within the context of existing and planned systems of systems?

3. Which are the most valuable capabilities that other systems can provide to my system if it becomes a part of existing and planned systems of systems?

4. To which systems of systems can my system contribute the most value?

5. Are there system of systems capabilities, behavior, and requirements that the system must address to become part of the existing and planned system of systems?

6. Am I designing my system so that it can be easily integrated with other systems?

7. Does my system have an adaptable and open architecture to enable future reconfiguration and integration into a system of systems?

8. Have the system of systems interface requirements been adequately defined and documented in the specification of my system?

9. Has my program developed and documented interface control requirements for external functional and physical interfaces?

10. Has my program identified and established conformance testing or certification mechanisms to assure that standards used by external interfaces conform to the prescribed interface specifications?

11. Has my program verified the external functional interface specifications to ensure that the functional and performance requirements for such interfaces are satisfied?

12. Does my system fully comply with external interface requirements identified through the JCIDS process and its accompanying documents and architectures (including the GIG architecture)?

13. Have I established rigorous interface design and management based on conformance and verification of standards at upper layers as well as at the application, transport, network, physical, media and data link communication layers?

A Contrasting Note about Engineering a Family of Systems. A family of systems is not considered to be a system per se. A family of systems does not create capability beyond the additive sum of the individual capabilities of its member systems. A family of systems is basically a grouping of systems having some common characteristic(s). For example, each system in a family of systems may belong to a domain or product lines (e.g., a family of missiles or aircraft). A family of systems lacks the synergy of a system of systems. The family of systems does not acquire qualitatively new properties as a result of the grouping. In fact, the member systems may not be connected into a whole.

4.3. Systems Engineering Activities in the System Life Cycle

DoD Instruction 5000.2 establishes the framework for acquisition programs. These programs are structured in phases, each separated by milestone decisions. In each phase of a system's life cycle, from concept to disposal, there are important systems engineering actions, which if properly performed, will assist the program manager in managing the program.

The purpose of this section is to acquaint program managers with the variety of acquisition documents that have systems engineering implications, either as sources of system parameters (e.g., the Initial Capabilities Document and Capability Development Document) or as the recipients of systems engineering analyses outputs (e.g., Acquisition Strategy, Analysis of Alternatives, etc.). This section shows how the systems engineering processes of Section 4.2 can be applied and tailored to each acquisition phase:

- Each phase builds upon the previous phase to further define the system technical solution;
- Systems engineering processes are iterated at each system element level; and
- Technical reviews serve to confirm outputs of the acquisition phases and major technical efforts within the acquisition phases.

As the by-phase discussions illustrate, there are a number of technical reviews appropriate to each acquisition phase that are conducted at all appropriate levels within a program. The purpose of these reviews is to provide the program manager with an integrated technical assessment of program technical risk and readiness to proceed to the next technical phase of the effort. Results of these reviews should be used to update the Systems Engineering Plan. Technical reviews should:

- Be event driven (vice schedule driven); conducted when the system under development satisfies review entry criteria as documented in the Systems Engineering Plan; and conducted, at a minimum, at the transition from one acquisition phase to the next and at major transition points of technical effort.
- Have their processes and requirements addressed in and required by contractual documents.

DoD Instruction 5000.2, Enclosure 3, presents the statutory, regulatory, and contract reporting information and milestone requirements for acquisition programs. These requirements are significant, and in some cases, the lead-time for preparation may exceed one year. The information and/or decisions that a program office reports in these documents often rely on analyses begun in pre-acquisition. During pre-acquisition, systems engineering processes translate user-defined capabilities into system specifications. As explained earlier, these systems engineering processes are both iterative and recursive. Likewise, some of the information requirements are iterative by milestone. Throughout this section, the terminology used to indicate a subsystem is either a system element, component, or configuration item, depending on the systems engineering context and phase of acquisition under discussion.

4.3.1. Concept Refinement Phase

Pre-acquisition, beginning with Concept Refinement, presents the first substantial opportunity to influence systems design by balancing technology opportunities, schedule constraints, funding availability, performance parameters, and operational requirements. Desired user capabilities, expressed in terms of Key Performance Parameters and other parameters, should be defined in terms of:

- Quantifiable metrics (e.g., speed, lethality) of performance to meet mission requirements affordably; and
- The full range of operational requirements (reliability, effectiveness, logistics footprint, supportability criteria, etc.) to sustain the mission over the long term.

Early and effective employment of systems engineering, applied in accordance with a well-structured Systems Engineering Plan, and monitored with meaningful systems engineering technical reviews, will reduce program risk and identify potential management issues in a timely manner.

The Concept Refinement phase refines the initial concept and generates a Technology Development Strategy. Entrance into this phase requires a successful Concept Decision and an approved Initial Capabilities Document. The Acquisition Decision Memorandum documents Milestone Decision Authority approval of the Analysis of Alternatives Plan and establishes a date for the Milestone A review. The Initial Capabilities Document and Analysis of Alternatives Plan guide Concept Refinement Phase activities.

4.3.1.1. Purpose of Systems Engineering in Concept Refinement

The Joint Capabilities Integration and Development System analysis process provides a structured methodology to identify capability gaps and needs, and suggest various approaches to provide needed capabilities within a specified functional or operational area. These analyses should incorporate innovative practices, including best commercial practices, collaborative environments, modeling and simulation, and electronic business solutions.

After the process identifies a materiel need, and an affirmative Concept Decision initiates Concept Refinement, the Analysis of Alternatives should use systems engineering processes to examine the alternatives and identify a preferred solution. Systems engineering processes can provide a technical evaluation of the operational effectiveness and estimated costs of the alternative system concepts that may provide a materiel solution to a needed mission capability. The analysis should assess the advantages and disadvantages of the alternatives under consideration, and include sensitivity analyses to possible changes in key assumptions or variables.

During Concept Refinement, systems engineering processes should also support development of the Technology Development Strategy for the preferred solution.

4.3.1.2. Inputs to the Systems Engineering Processes in Concept Refinement

The following information sources provide important inputs to the systems engineering processes supporting Concept Refinement:

- Initial Capabilities Document;
- Analysis of Alternatives Plan;

- Exit Criteria for the Concept Refinement Phase; and
- Alternative Maintenance and Logistics Concepts.

4.3.1.3. Key Systems Engineering Activities During Concept Refinement

Figure 1 identifies the systems engineering-related steps during the Concept Refinement Phase. Paragraphs below contain additional detail on each step.

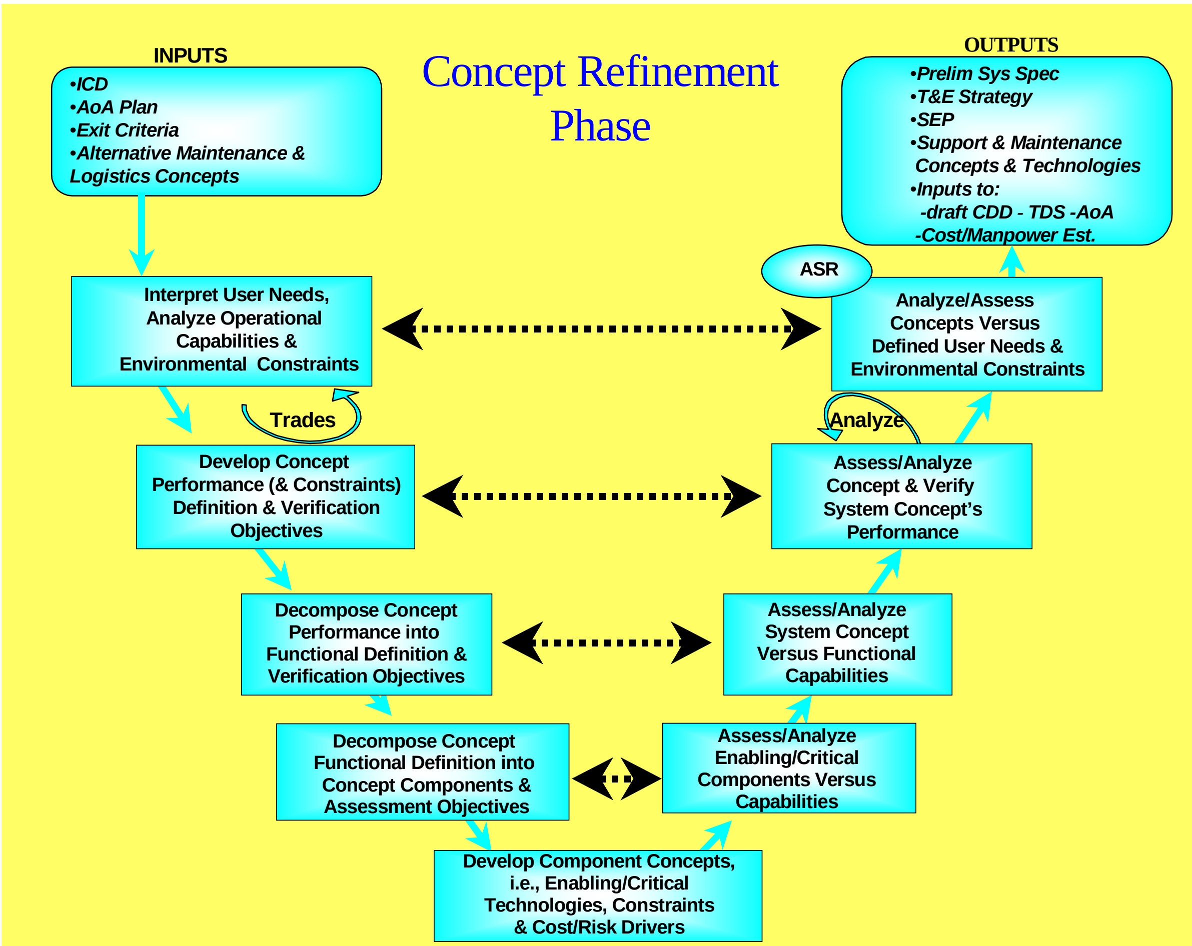

Figure 1. Systems engineering-related steps during Concept Refinement

4.3.1.3.1. Interpret User Needs; Analyze Operational Capabilities and Environmental Constraints

This step includes the aggregation of all inputs available at this stage of the program (Initial Capabilities Document, Analysis of Alternatives Plan, exit criteria for the phase, concept alternatives for overall tactical system, as well as associated support system, training system, and interoperable systems). Further analysis and definition is typically required to ascertain all of the related constraints to be applied to the effort:

- Environmental—systems threats, usage environment, support environment, doctrine, operational concepts;
- Resource—industrial base; notional available development, operation, and support budgets; required date for system fielding;
- Technology—applicable technology base to be used for concept maturation; and
- Statutory and regulatory—the Federal Acquisition Regulation; the DoD 5000-series; etc.

Key to this initial step of concept refinement is to ensure that all drivers of the concept definition are completely captured and managed as an integrated whole, and that all of the drivers can be met by each of the concept alternatives under consideration. This defines the expectations of the overall system concept, and defines the trade space and risk associated with each of the constraints, above. Defining the trade space and risk enables the comprehensive analysis of system alternatives, and allows a rational selection of a preferred system concept. The preferred system concept should strike the best balance in providing the needed capabilities within the constraints on the program.

4.3.1.3.2. Develop Concept Performance (and Constraints) Definition and Verification Objectives

This step includes the analysis and decomposition (from capability level to system level) of system performance and system design constraints traceable back to those capabilities and constraints defined in Section 4.3.1.3.1 above. All capabilities and environmental constraints should be decomposed to the system performance level. They should be re-analyzed to determine the extent to which alternative concepts can meet all capability needs within program constraints (as needs and constraints become better understood as a result of decomposition). The trade space and risk should be analyzed and assessed for each alternative concept. For each alternative system concept, expected performance capabilities should be explicitly defined and related to the capability needs. To the extent concept performance can only be met through trade offs (due to incompatibility of capabilities/constraints), changes may be required to the capability or constraints previously defined.

Verification planning should define the test requirements needed to evaluate the ability of the matured system concept(s) to meet requirements.

4.3.1.3.3. Decompose Concept Performance into Functional Definition and Verification Objectives

This step includes the further decomposition of concept system performance to the functional level. Consideration should be given to inclusion of functionality and functional flow definition across the full system concept (tactical system, support system, training system) and how this functionality relates to other interoperable systems (functional interfaces). Critical to this analysis is an understanding of the level of functionality achievable within program constraints and risk. Trade space and risk should be analyzed and assessed against desired functional performance. Trade offs are made to stay within program constraints and may require changes to higher-level system or concept definitions.

System functional verification planning should enable test and evaluation of the matured system concept functionality.

4.3.1.3.4. Decompose Concept Functional Definition into Concept Components and Assessment Objectives

This step includes the allocation of concept functions into components of the concept that will execute the functionality. Critical to this analysis is an understanding of what functional performance is enabled by multiple systems, or system components, operating as a functional entity. Hardware elements, software elements, physical interfaces, functional interfaces, standards, existing, and to-be-developed elements, should all be considered and defined in the concept. As in previous steps, this level of decomposition and allocation may induce trades to stay within program constraints. These trades need to be reflected in higher level functional, system, and capability definitions, which should be updated accordingly.

Concept component verification planning should enable testing and validation of critical concept components.

4.3.1.3.5. Develop Component Concepts, Including Enabling/Critical Technologies, Constraints, and Cost/Risk Drivers

At this point, all of the basic concept design requirements should have been analyzed, defined, and reconciled with constraints. The system concept(s) components should have been synthesized and substantiated (e.g., through analyses, modeling and simulation, demonstrations, etc.) to allow verification of components against requirements, and integration of the components into an overall system for further verification and validation. Key to this step is the development of conceptual components to demonstrate the viability of the overall concept, indicate where additional technology maturation should occur, and validate that acceptable trade space between expected capabilities and program constraints exists to accomodatepotential risk.

4.3.1.3.6. Analyze and Assess Enabling/Critical Components Versus Capabilities

Utilizing the component verification plans developed as part of the functional allocation, the enabling and/or critical components of the concept should be evaluated. Evaluation results should be assessed against component requirements and the impact on the overall concept capabilities and constraints determined. Critical to this step is the understanding of test results and how the concept component functionality verifies or contradicts the desired capabilities, as well as what component technologies are required and the level of achievable performance. Capability trade offs within the available trade space, or further component concept development within program and concept constraints may be required.

4.3.1.3.7. Analyze and Assess System Concept Versus Functional Capabilities

Utilizing the concept functional verification plans developed as part of the functional analysis and decomposition, overall system functionality should be evaluated. Concept components should be integrated and assessed from a functional standpoint relative to desired capabilities. Critical to this step is understanding how the enabling components work together as an integrated whole to provide functionality at the component and system levels, and how the achieved functionality relates to the overall desired capability. Also important is an understanding of the technology development required to achieve critical functions. Capability trade offs within the available trade space, or further refinement of functionality within program and concept constraints may be required.

4.3.1.3.8. Analyze and Assess Concept and Verify System Concept's Performance

Utilizing the verification objectives previously defined, evaluate the overall integrated concept against system performance objectives and constraints. Concept components are integrated from both physical and functional perspectives across the full concept domain (tactical, support, training, etc.). Critical to this step is an understanding of overall system concept capability versus need, level of achievable performance within the complete set of constraints, and the enabling technologies requiring further development. Trades at this level will include decisions as to acceptable technology risk versus desired performance.

4.3.1.3.9. Analyze and Assess Concepts Versus Defined User Needs and Specified Environmental Constraints

Based upon the results of the verification of components, functionality, and system performance, a determination of the preferred system concept should be made. Advantages and disadvantages of various approaches should be documented and included in the analysis of alternatives. Trade offs of achievable performance should be complete and captured in a preliminary system specification. Enabling technologies requiring further development to achieve acceptable levels of risk should be defined and plans should be developed for technology development. The preliminary system specification serves as the guiding technical requirement for this development effort.

4.3.1.4. Technical Reviews during Concept Refinement

4.3.1.4.1. Initial Technical Review (ITR)

The ITR is a multi-disciplined technical review to support a program's initial Program Objective Memorandum submission. This review ensures that a program's technical baseline is sufficiently rigorous to support a valid cost estimate (with acceptable cost risk), and enable an independent assessment of that estimate by cost, technical, and program management subject matter experts. The ITR assesses the capability needs and conceptual approach of a proposed program and verifies that the requisite research, development, test, engineering, logistics, and programmatic bases for the program reflect the complete spectrum of technical challenges and risks. Additionally, the ITR ensures that historical and prospective drivers of system cost have been quantified to the maximum extent and that the range of uncertainty in these parameters has been captured and reflected in the program cost estimates.

Per DoD Instruction 5000.2, the program manager for Acquisition Category I and IA programs must define program and system parameters in a Cost Analysis Requirements Description (CARD), as described in DoD 5000.4M. The basic CARD technical and programmatic guidance, tailored to suit the scope and complexity of the program, should be followed to ensure that all pertinent technical cost drivers are addressed. The success of the ITR also depends on independent subject matter expert review of each of the identified cost drivers. The subject matter experts should be drawn from the correct technical competencies that specialize in each of the areas addressed in a CARD-like document, and the cost drivers detailed in the CARD-like document should be used properly in the development of the program cost estimate. Completion of the ITR should provide:

(1) A complete CARD-like document detailing system overview, risk, and system operational concept;

(2) An assessment of the technical and cost risks of the proposed program; and

(3) An independent assessment of the program's cost estimate.

Typical ITR success criteria include affirmative answers to the following exit questions:

(1) Does the CARD-like document capture the key program cost drivers, development costs (all aspects of hardware, human integration, and software), production costs, operation and support costs? Is the CARD-like document complete and thorough?

(2) Are the underlying assumptions used in developing the CARD-like document technically and programmatically sound and complete?

(3) Have the appropriate technical and programmatic competencies been involved in the CARD-like document development, and have the proper subject matter experts been involved in its review?

(4) Are the risks known and manageable within the cost estimate?

(5) Is the program, as captured in the CARD-like document, executable?

4.3.1.4.2. Alternative System Review (ASR)

The ASR is a multi-disciplined technical review to ensure that the resulting set of requirements agrees with the customers' needs and expectations and that the system under review can proceed into the Technology Development phase. The ASR should be complete prior to Milestone A. Generally this review assesses the alternative systems that have been evaluated during the Concept Refinement phase, and ensures that the preferred system alternative is cost effective, affordable, operationally effective and suitable, and can be developed to provide a timely solution to a need at an acceptable level of risk. Of critical importance to this review is the understanding of available system concepts to meet the capabilities described in the Initial Capabilities Document and the affordability, operational effectiveness, and technology risks inherent in each alternative concept. Depending on the overall acquisition strategy, one or more preferred solutions may carry forward into the Technology Development phase.

By reviewing alternative system concepts, the ASR helps ensure that sufficient effort has been given to conducting trade studies that consider and incorporate alternative system designs that may more effectively and efficiently meet the defined capabilities. A successful review is predicated on the IPT's determination that the operational capabilities, preferred solution(s), available technologies, and program resources (funding, schedule, staffing, and processes) form a satisfactory basis for proceeding into the Technology Development phase. The program manager should tailor the review to the technical scope and risk of the system, and address the ASR in the Systems Engineering Plan.

Completion of the ASR should provide:

(1) An agreement on the preferred system concept(s) to take forward into Technology Development.

(2) Software architectural constraints/drivers to address Defense Information Infrastructure / Common Operating Environment and system extensibility requirements.

(3) An assessment of the full system software concept to include conceptual definition of the complete deliverable/non-deliverable software, scope, and risk (e.g., operational software elements, software engineering environment, test software, maintenance software, simulation/stimulation software, training software, in-service support software, etc.).

(4) A comprehensive rationale for the preferred solution, including the Analysis of Alternatives that evaluated relative cost, schedule, performance (hardware, human, software), and technology risks.

(5) A comprehensive assessment of the relative risks associated with including Commercial Off-the-Shelf or Non-Developmental Items in the program, with emphasis on host platform environmental design, diagnostic information integration, and maintenance concept compatibility.

(6) A comprehensive risk assessment for the Technology Development phase.

(7) Trade studies/technical demonstrations for concept risk reduction.

(8) Joint requirements for the purposes of compatibility, interoperability, and integration.

(9) Refined thresholds and objectives initially stated as broad measures of effectiveness.

(10) Completed, comprehensive planning for the Technology Development phase (hardware and software), that addresses critical components to be developed and demonstrated, their cost, and critical path drivers.

(11) Initial planning for the System Development and Demonstration phase.

(12) A draft system requirements document if one does not already exist. (This is a high-level engineering document that represents the customer/user capability needs as system requirements). This systems requirement document should include a system level description of all software elements required by the preferred system concept.

The ASR is important because it is a comprehensive attempt to ensure that the system requirements are aligned with the customer's needs. The ASR attempts to minimize the number of requirements that may need to be changed in later phases. Changing requirements later in the program will usually entail cost increases and scheduling slips.

Typical ASR success criteria include affirmative answers to the following exit questions:

(1) Can the preferred solution(s) satisfy the Initial Capabilities Document?

(2) Is the preferred solution(s) sufficiently detailed and understood to enable entry into Technology Development with low technical risk?

(3) Is the system software scope and complexity sufficiently understood and addressed in the planning for the Technology Development phase to enable low software technical risk?

(4) Are the risks for Technology Development known and manageable?

(5) Is the program schedule executable (technical/cost risks)?

(6) Is the program properly staffed?

(7) Is the Technology Development work effort executable within the existing budget?

(8) Has a preliminary system specification, consistent with technology maturity and the proposed program cost and schedule, captured the system technical baseline?

4.3.1.4.3. Summary of Outputs of the Systems Engineering Processes in Concept Refinement

- Preliminary System Specification;
- T&E Strategy;
- Systems Engineering Plan;
- Support and Maintenance Concepts and Technologies;
- Inputs to draft Capability Development Document;
- Inputs to Technology Development Strategy;
- Inputs to Analysis of Alternatives;
- Inputs to Cost and Manpower Estimate.

4.3.2. Technology Development Phase

A successful Milestone A decision initiates the Technology Development phase. Per DoD Instruction 5000.2, this phase reduces technology risk and determines the appropriate set of technologies to be integrated into a full system. Technology development is a continuous technology discovery and development process that reflects close collaboration between the Science and Technology community, the user, and the developer. Technology development is an iterative process of assessing technologies and refining user performance parameters. The Initial Capabilities Document, the Technology Development Strategy, and working the draft Capability Development Document guide the phase efforts, leading to the Capability Development Document.

4.3.2.1. Purpose of Systems Engineering in Technology Development

During Technology Development, systems engineering provides comprehensive, iterative processes to accomplish the following activities:

- Convert each required capability into a system performance specification;
- Translate user-defined performance parameters into configured systems;
- Integrate the technical inputs of the entire design team;
- Manage interfaces;
- Characterize and manage technical risk;
- Transition technology from the technology base into program specific efforts; and
- Verify that designs meet operational needs.

Systems engineering processes develop the suite of technologies for the preferred system solution.

4.3.2.2. Inputs to the Systems Engineering Processes in Technology Development

The following information sources provide important inputs to the systems engineering processes supporting Technology Development:

- Initial Capabilities Document and draft Capability Development Document;
- Preferred System Concept;
- Exit Criteria;
- Test and Evaluation Strategy;
- Support and Maintenance Concepts and Technologies;
- Analysis of Alternatives;
- Systems Engineering Plan; and
- Technology Development Strategy.

4.3.2.3. Key Systems Engineering Activities During Technology Development

Figure 16 identifies the systems engineering-related steps during the Technology Development Phase. Paragraphs 4.3.2.3.1 through 4.3.2.4.3 contain additional detail on each step.

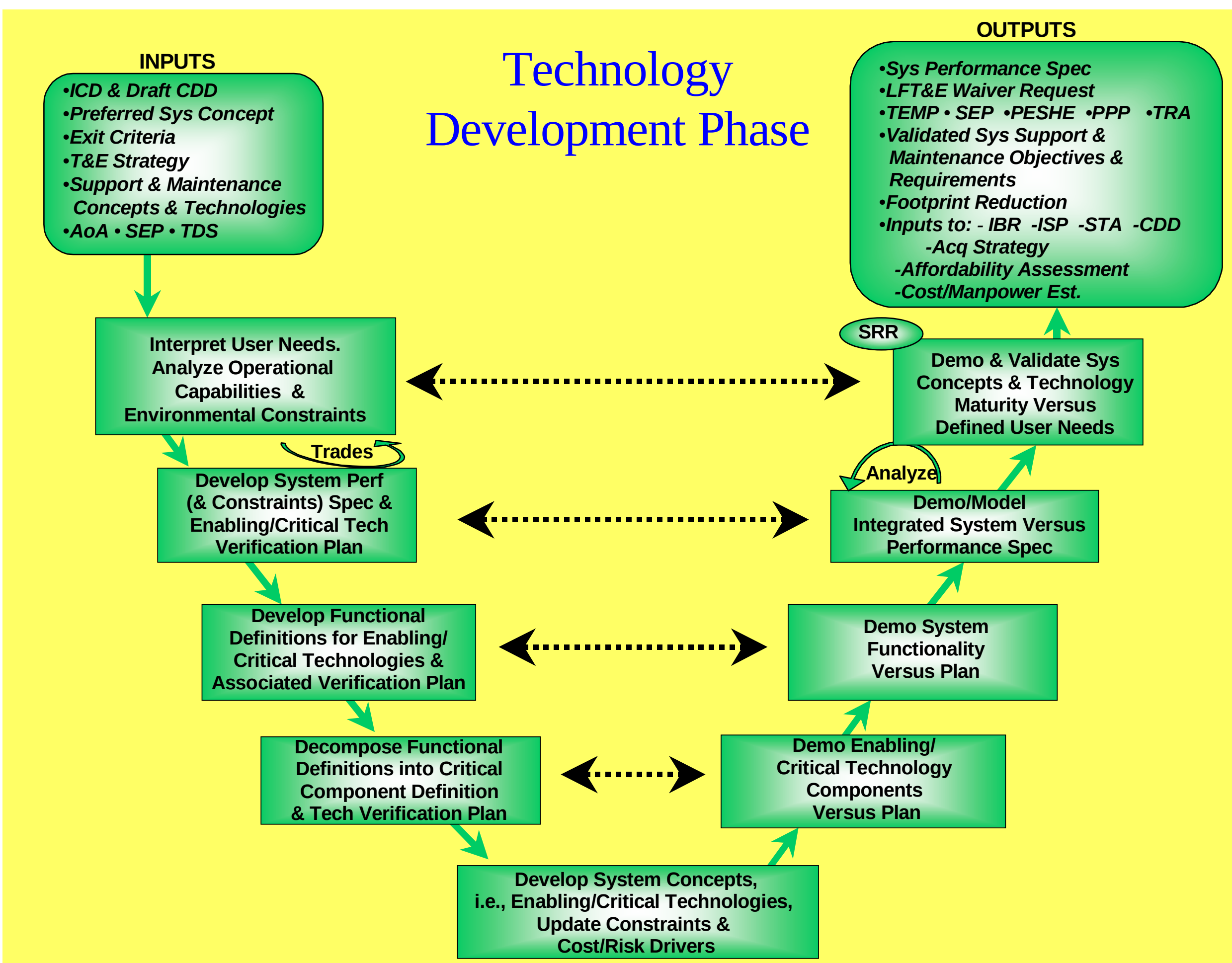

Figure 2. Systems engineering-related steps during Technology Development

4.3.2.3.1. Interpret User Needs; Analyze Operational Capabilities and Environmental Constraints

This step includes the aggregation of all inputs available at this stage of the program (Initial Capabilities Document, draft Capability Development Document, results of the Analysis of Alternatives and identification of the preferred system concept, exit criteria for the phase, Systems Engineering Plan, Technology Development Strategy, Test and Evaluation Strategy, as well as associated support and maintenance concepts and technologies, training system, and interoperable systems). Additional analysis and definition may be required to ascertain all of the related constraints to be applied to the effort:

- Environmental—systems threats, usage environment, support environment, doctrine, operational concepts, etc.;
- Resource—industrial base; notional available development, operation, and support budgets; and the required date for system fielding;
- Technology—applicable technology base to be used for technology development; and
- Statutory and regulatory—the Federal Acquisition Regulation; the DoD 5000-series; etc.

Key to this technology development effort is ensuring that all aspects of the required technology are adequately matured and managed as an integrated whole, and can support the user needs via the preferred concept. This not only ensures that overall expectations are explicitly defined, but that trade space and risk in each of the areas above are defined to enable comprehensive analysis of technology availability and rational formulation of a system performance specification that strikes the best balance in meeting all of the needed capabilities within the many constraints on the program.

4.3.2.3.2. Develop System Performance (and Constraints) Specifications and Enabling/Critical Technologies Verification Plan

This step includes the further analysis and decomposition (from capability level to system level) of system performance and system design constraints, traceable back to those capabilities and constraints defined above. All capabilities and environmental constraints should be decomposed to the system performance level. They should be re-analyzed to determine the extent to which available technologies can meet the full spectrum of needs and constraints (as needs and constraints become better understood as a result of decomposition). The trade space and risk should be analyzed and assessed against available technologies. The enabling and/or critical technologies should be identified. Each technology performance capability should be explicitly defined and related to the capability needs. To the extent performance can only be met through trade offs of certain aspects (due to incompatibility of capabilities/constraints), changes may be required to the capability or constraints previously defined.

Verification planning should define the test requirements needed to evaluate the ability of enabling and/or critical technologies to meet system requirements.

4.3.2.3.3. Develop Functional Definitions for Enabling/Critical Technologies and Associated Verification Plan

This step requires the further decomposition of system performance to the functional level. The functional requirements should be evaluated against available technologies, such that enabling and/or critical technologies can be defined. Consideration should be given to inclusion of functionality and functional flow definition across the full system (tactical system, support system, training system) and how this functionality relates to other interoperable systems (functional interfaces). Critical to this analysis is an understanding of the level of functionality achievable within the program constraints and program risk. Trade space and risk should be analyzed and assessed against desired functional performance. Trade offs may be required to stay within program constraints and may require changes to higher-level system definitions.

System functional verification planning should develop the test requirements to evaluate system functionality and the maturity of the enabling/critical technologies.

4.3.2.3.4. Decompose Functional Definitions into Critical Component Definition and Technology Verification Plan

This step includes the allocation of system functions into critical components of the system that will provide the required functionality. Key to this analysis is an understanding of what functional performance is enabled by multiple systems, or system components, operating as a functional entity. Hardware elements, software elements, physical interfaces, functional interfaces, standards, existing and to-be-developed technology elements, should all be considered and defined in the system specification. As in previous steps, this level of decomposition and allocation may induce trades to stay within program constraints. These trades should be reflected in higher level functional, system, capability definitions, and system specifications (i.e., these engineering entities should be updated accordingly).

System component verification planning should enable testing and validation of critical system components.

4.3.2.3.5. Develop System Concepts, i.e., Enabling/Critical Technologies; Update Constraints and Cost/Risk Drivers

At this point, all of the basic system design requirements should have been analyzed, defined, and reconciled with constraints. The system components are synthesized and substantiated (e.g., through analyses, modeling and simulation, demonstrations, etc.) to allow verification of the components against requirements, and integration of the components into an overall system for further validation. Key to this step is the development of system concepts that will demonstrate the viability of the overall system, indicate where enabling and/or critical technology maturation should occur, and validation that acceptable trade space and risk exists within the program constraints.

4.3.2.3.6. Demonstrate Enabling/Critical Technology Components Versus Plan

Using the system component verification planning developed as part of the functional allocation, the system enabling/critical technology components should be evaluated. Evaluation results should be assessed against system component requirements, and the impact on the overall system capabilities and constraints determined. Critical to this step is the understanding of test results and how the system component functionality verifies or contradicts the desired capabilities, as well as what enabling and/or critical component technologies are required and the level of achievable performance. Trade offs to system capability or additional system component development may be required, within the program and system constraints and trade space available.

4.3.2.3.7. Demonstrate System Functionality Versus Plan

Utilizing the system functional verification plans developed as part of the functional analysis and decomposition, the overall system functionality should be evaluated. System components are integrated and assessed from a functional standpoint relative to desired capabilities. Critical to this step is the understanding of how the enabling components work together as an integrated whole to enable functionality at the system level, and how the achieved functionality relates to the overall desired system capability. Also important is an understanding of the enabling and/or critical technology maturity required to achieve critical functions. Trade offs of desired capability, or further refinement of functionality may be required within program and system constraints, and available trade space.

4.3.2.3.8. Demonstrate/Model the Integrated System Versus the Performance Specification

Utilizing Engineering Development Models (EDMs), modeling and simulation, and the verification objectives previously defined (section 4.3.2.3.2.), evaluate the overall integrated system against system performance objectives and constraints. System components are integrated from both physical and functional perspectives across the full system domain (tactical, support, training, etc.). Critical to this step is an understanding of: overall system capability versus need, level of achievable performance within the complete set of constraints, and the enabling/critical technologies requiring further development. Trades at this level will include decisions as to acceptable technology risk versus desired system performance.

4.3.2.3.9. Demonstrate and Validate the System Concepts and Technology Maturity Versus Defined User Needs

Based upon the results of the verification of components, functionality, and system performance, a System Performance Specification should be created. Trade-offs of achievable performance should be complete and captured in the Systems Specification. Critical and/or enabling technologies should have demonstrated adequate maturity to achieve acceptable levels of risk. The System Performance Specification serves as the guiding technical requirement for the system development effort.

4.3.2.4. Technical Reviews during Technology Development

4.3.2.4.1. System Requirements Review (SRR)

The SRR is conducted to ascertain progress in defining system technical requirements. This review determines the direction and progress of the systems engineering effort and the degree of convergence upon a balanced and complete configuration. It is normally held during Technology Development, but may be repeated after the start of System Development and Demonstration to clarify the contractor's understanding of redefined or new user requirements.

The SRR is a multi-disciplined technical review to ensure that the system under review can proceed into the System Development and Demonstration phase, and that all system requirements and performance requirements derived from the Initial Capabilities Document or draft Capability Development Document are defined and are consistent with cost (program budget), schedule (program schedule), risk, and other system constraints. Generally this review assesses the system requirements as captured in the system specification, and ensures that the system requirements are consistent with the preferred system solution as well as available technologies resulting from the Technology Development phase. Of critical importance to this review is an understanding of the program technical risk inherent in the system specification and in the System Development and Demonstration Phase Systems Engineering Plan. Determining an acceptable level of risk is key to a successful review.

Completion of the SRR should provide:

(1) An approved preliminary system performance specification;

(2) A preliminary allocation of system requirements to hardware, human, and software subsystems;

(3) Identification of all software components (tactical, support, deliverable, non-deliverable, etc.);

(4) A comprehensive risk assessment for System Development and Demonstration;

(5) An approved System Development and Demonstration Phase Systems Engineering Plan that addresses cost and critical path drivers; and

(6) An approved Product Support Plan with updates applicable to this phase.

During the SRR, the systems requirements are evaluated to determine whether they are fully defined and consistent with the mature technology solution, and whether traceability of systems requirements to the Initial Capabilities Document or draft Capability Development Document is maintained. A successful review is predicated on the IPT's determination that the system requirements, preferred system solution, available technology, and program resources (funding, schedule, staffing, and processes) form a satisfactory basis for proceeding into the SDD phase. The program manager should tailor the review to the technical scope and risk of the system, and address the SRR in the Systems Engineering Plan.

Typical SRR success criteria include affirmative answers to the following exit questions:

(1) Can the system requirements, as disclosed, satisfy the ICD or draft CDD?

(2) Are the system requirements sufficiently detailed and understood to enable system functional definition and functional decomposition?

(3) Is there an approved system performance specification?

(4) Are adequate processes and metrics in place for the program to succeed?

(5) Have Human Systems Integration requirements been reviewed and included in the overall system design?

(6) Are the risks known and manageable for development?

(7) Is the program schedule executable (technical and/or cost risks)?

(8) Is the program properly staffed?

(9) Is the program executable within the existing budget?

(10) Does the updated cost estimate fit within the existing budget?

(11) Is the preliminary Cost Analysis Requirements Description consistent with the approved system performance specification?

(12) Is the software functionality in the system specification consistent with the software sizing estimates and the resource-loaded schedule?

(13) Did the Technology Development phase sufficiently reduce development risks?

The SRR is important in understanding the system performance, cost, and scheduling impacts that the defined requirements will have on the system. This is the last dedicated review of the system requirements, unless an additional SRR is held after the refining of the system performance constraints during the System Development and Demonstration Phase (see Section 5.3.3.).

4.3.2.4.2. Integrated Baseline Review (IBR)

Program managers should use the IBR throughout the program when Earned Value Management is required. This review has a business focus, but should include the important technical considerations discussed below. The process is composed of four steps:

(1) The Program Manager’s assessment of their understanding of the risks;

(2) Preparation for an IBR;

(3) Execution of the IBR; and

(4) The management process (the source of on-going mutual understanding).

The key step in the process is execution of the IBR. The IBR establishes a mutual understanding of the project Performance Measurement Baseline (PMB). This understanding provides for an agreement on a plan of action to evaluate the risks inherent in the PMB and the management processes that operate during project execution. Completion of the review should result in the assessment of risk within the PMB and the degree to which the following have been established:

(1) Technical scope of work is fully included and is consistent with authorizing documents;

(2) Key project schedule milestones are identified and supporting schedules reflect a logical flow to accomplish the work;

(3) Resources (budgets, facilities, personnel, skills, etc.) are available and are adequate for the assigned tasks;

(4) Tasks are planned and can be measured objectively relative to the technical progress;

(5) Rationales underlying the PMB are reasonable; and

(6) Management processes support successful execution of the project.

Section 11.3.4 describes an IBR in detail. The Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics, in cooperation with industry, has also prepared an IBR handbook.

4.3.2.4.3. Technology Readiness Assessment (TRA)

Per DoD Instruction 5000.2, the TRA is a regulatory information requirement for all acquisition programs. The TRA is a systematic, metrics-based process that assesses the maturity of Critical Technology Elements. The TRA may be conducted concurrently with other Technical Reviews, specifically the System Requirements Review, the Critical Design Review, the System Verification Review, or the Production Readiness Review. If a platform or system depends on specific technologies to meet system operational threshold requirements in development, production, and operation, and if the technology or its application is either new or novel, then that technology is considered a Critical Technology Element. The TRA should not be considered a *risk* assessment, but it should be viewed as a tool for assessing program risk and the adequacy of technology maturation planning. The TRA scores the current readiness level of selected system elements, using defined Technology Readiness Levels. The TRA highlights critical technologies and other potential technology risk areas that require program manager attention. The TRA essentially “draws a line in the sand” on the day of the event for making an assessment of technology readiness for critical technologies integrated at some elemental level. If the system does not meet pre-defined Technology Readiness Level scores, then a Critical Technology Element maturation plan is identified. This plan explains in detail how the Technology Readiness Level will be reached prior to the next milestone decision date or relevant decision point. Completion of the TRA should provide:

(1) A comprehensive review, using an established program Work Breakdown Structure as an outline, of the entire platform or system. This review, using a conceptual or established baseline design configuration, identifies program Critical Technology Elements;

(2) An objective scoring of the level of technological maturity for each Critical Technology Element by subject matter experts;

(3) Maturation plans for achieving an acceptable maturity roadmap for Critical Technology Elements prior to critical milestone decision dates; and

(4) A final report documenting the findings of the assessment panel.

After the final report is written, the chairman submits the report to the appropriate Service officials and the program manager. Once approved, the report and cover letter are forwarded to the service acquisition official. For ACAT ID or IAM programs, the service acquisition official provides a recommendation to DDR&E for DUSD(S&T) final approval. If deemed necessary, the DDR&E can conduct an Independent Technical Assessment (ITA) in addition to, and totally separate from, the program TRA.

4.3.2.5. Outputs of the Systems Engineering Processes in Technology Development

- Preliminary System Performance Specification;
- Live-Fire T&E Waiver request;
- Test and Evaluation Master Plan;
- Systems Engineering Plan;
- Programmatic Environment, Safety, and Occupational Health Evaluation (PESHE);
- Program Protection Plan;
- Technology Readiness Assessment;
- Validated System Support and Maintenance Objectives and Requirements (add this link here: http://acc.dau.mil/simplify/file_download.php/FINAL+GUIDE+with+Memo+-+October+24.pdf?URL_ID=15943&filename=10772113271FINAL_GUIDE_with_Memo_-_October_24.pdf&filetype=application%2Fpdf&filesize=432407&name=FINAL+GUIDE+with+Memo+-+October+24.pdf&location=user-S/#page=21);
- Footprint Reduction;
- Inputs to the Integrated Baseline Review;
- Inputs to the Information Support Plan;
- Inputs to the System Threat Assessment;
- Inputs to the Capability Development Document;
- Inputs to the Acquisition Strategy;
- Inputs to the Affordability Assessment; and
- Inputs to the Cost and Manpower Estimate.

4.3.3. System Development and Demonstration Phase

A program usually enters the acquisition process at Milestone B, when the Milestone Decision Authority permits the system to enter the System Development and Demonstration phase and initiates the program. A key emphasis during System Development and Demonstration is to ensure operational supportability with particular attention to minimizing the logistics footprint.

The purposes of System Development and Demonstration are to:

- Develop a system or increment of capability;
- Reduce integration and manufacturing risk;
- Ensure operational supportability with particular attention to reducing the logistics footprint;
- Implement human systems integration;
- Design for producibility;

- Ensure affordability and protection of critical program information; and
- Demonstrate system integration, interoperability, safety, and utility.

In System Development and Demonstration, the program, the system architecture, and system elements down to the configuration item level are defined based upon the mature technology suite selected and integrated during Concept Refinement and Technology Development. During System Development and Demonstration, system design requirements are allocated down to the major subsystem level, and are refined as a result of developmental and operational tests, and iterative systems engineering analyses. The support concept and strategy are refined.

Two work efforts, separated by the Design Readiness Review, comprise System Development and Demonstration: System Integration and System Demonstration.

4.3.3.1. Inputs to the Systems Engineering Processes in System Integration

Inputs to the Systems Engineering processes in System Development and Demonstration include the following:

- System Performance Specification;
- Exit Criteria;
- Validated System Support and Maintenance Objectives and Requirements;
- Acquisition Program Baseline;
- Capability Development Document;
- Systems Engineering Plan;
- Information Support Plan;
- Test and Evaluation Master Plan; and
- Product Support Strategy.

4.3.3.2. Purpose of Systems Engineering in System Integration

The System Integration work effort begins when the program manager has a technical solution for the system or increment of capability, but has not integrated the components and subsystems into a system. Through the use of systems engineering, the System Integration effort integrates components and subsystems, completes the detailed design, and reduces system level risk. The effort typically includes the demonstration of prototype articles or engineering development models.

4.3.3.3. Key Systems Engineering Activities During System Integration

Figure 17 identifies the systems engineering-related steps during the System Integration effort of the System Development and Demonstration Phase. Paragraphs 4.3.3.3.1 through 4.3.3.4.5. contain additional detail on each step.

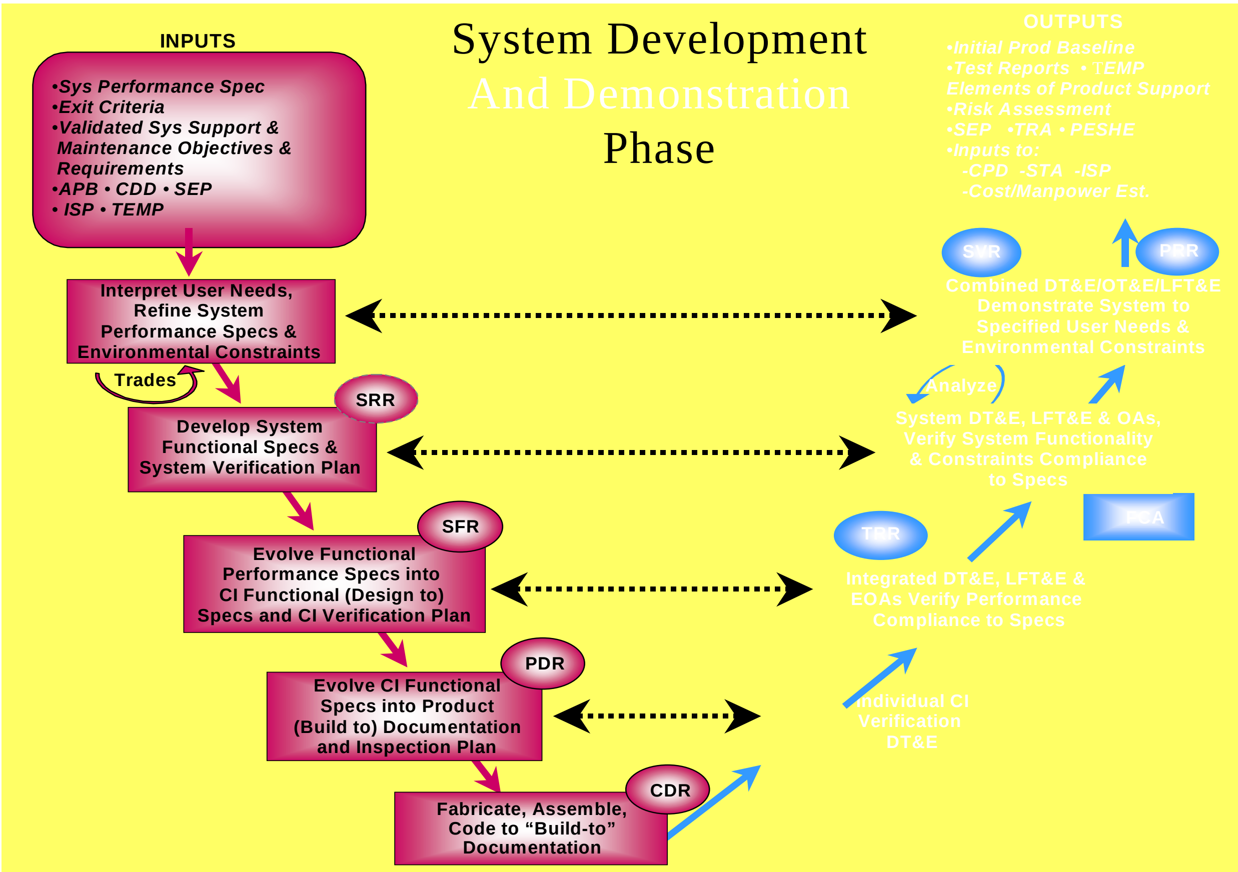

Figure 3. Systems engineering-related steps during the System Integration effort of System Development and Demonstration.

4.3.3.3.1. Interpret User Needs, Refine System Performance Specifications and Environmental Constraints

This step includes understanding all of the inputs available at this stage of the program, including the Initial Capabilities Document, Capability Development Document, Acquisition Program Baseline, Systems Engineering Plan, Test and Evaluation Master Plan, as well as validated system support and maintenance concepts and technologies. The users and the requirements authority have already approved a minimum set of key performance parameters that are included in the Capability Development Document that guides the efforts of this phase. As the design matures, the program manager may conduct trade studies on the threshold and objective levels, and refine the key performance parameters thresholds and objectives with the approval of the requirements authority.

Throughout the development activities, the program manager should maintain a thorough understanding of the key performance parameters, other specified performance parameters, and the suite of matured technologies resulting from the Technology Development phase. The program manager should ensure that all aspects of the specified system are adequately matured and managed as an integrated whole. The refined system specifications should consider all life-cycle processes and constraints, such as system availability, supportability, logistics footprint, training, and other logistics requirements, developmental and operational test environments and scenarios, and disposal. For example, the program manager should plan the Environment, Safety, and Occupational Health assessment. The program manager should develop and manage the system requirements stemming from the life-cycle considerations, and use prototypes to ensure user and other stakeholder buy-in as the design matures. The program manager should continually update cost and schedule estimates synchronized with the Systems Engineering Plan and Program Plan. The program manager should continually address and characterize technical risk, and prepare for an additional System Requirements Review, if required.

4.3.3.3.2. Develop System Functional Specifications and System Verification Plan

This step determines the required system functions based on the Capability Development Document performance parameters and all other requirements and constraints, and allocates subsystems to each function. Partitioning of the system into subsystems leads to the definition of subsystem interfaces and integration requirements. The engineers define hardware, human, and software functional expectations, and establish the system functional baseline for the System Functional Review that follows this step. The program manager should continually monitor system cost, schedule, and risk. The program manager should factor all design considerations into trade studies, and incorporate them into the design. The program manager should develop plans for the subsystem integration, verification, and validation processes, as well as verification and validation plans for the system as a whole. The planning should consider all interface functional and performance specifications.

4.3.3.3.3. Evolve Functional Performance Specifications into Configuration Item (CI) Functional ("Design-to") Specifications and CI Verification Plan

This step involves allocating functional performance specifications into system functional and performance requirements allocated across the CIs. Enabling or critical technologies, the envisioned operational environment(s), the "ilities," and the other logistics elements should be part of satisfying performance needs. The program manager should plan to test or verify the configuration items for functionality and performance. The program manager should continually monitor risk and assess its impact on cost, schedule, and performance. Additional analyses conducted at this step include a Failure Mode Effects and Criticality Analysis, a Failure Tree Analysis, and a Reliability-Centered Maintenance (RCM) Analysis.

The program manager should convene a Preliminary Design Review after this step and approve the allocated baseline. The allocated baseline includes all functional and interface characteristics allocated from the system, interface requirements with other CIs, and design constraints. The allocated baseline should describe the verification required to demonstrate the achievement of specified functional and interface characteristics.

4.3.3.3.4. Evolve CI Functional Specifications into Product ("Build-to") Documentation and Inspection Plan

This step finalizes the detailed design of the system. The design should include all hardware and software components. The engineers should complete drawings and other documentation for "building" the components (i.e., fabricating hardware components or coding the software element) and plan for the integration and testing of all of the components. The program manager should plan the acquisition of any commercial item components or reuse of components from some other effort. Environment, Safety and Occupational Health and other life-cycle and/or environmental considerations that affect the component level of the system should be part of the decision-making and trade studies that occur at this level of design. The program manager should continually assess cost, schedule, and performance. Additional analyses at this step include a Level of Repair Analysis and a Maintenance Task Analysis. Analysts should estimate the projected system reliability from demonstrated reliability rates.

The program manager should convene a Critical Design Review at the end of this step. The end product of the Critical Design Review is a product baseline. The majority of production capable system drawings should have been validated and approved prior to the Critical Design Review.

4.3.3.3.5. Fabricate, Assemble, Code to "Build-to" Documentation

This step involves fabricating hardware components and coding software components; acquiring all other components, including commercial items, being bought or reused; and then assembling the components according to the integration (and test) planning. At this point, all the system, subsystem, and component design requirements should have been developed. The program manager should manage the design requirements and plan for corrective action for any discovered hardware and software deficiencies. If any technology is not mature enough to be used in the current increment, the program manager should integrate and test an alternative, mature, technology in its place. The program manager should relegate the immature technology to the next increment of the system. The program manager should continually assess cost, schedule, and performance.

This step will usually result in prototypes and engineering development models, and should include developmental testing to support the Design Readiness Review. During this time, the program manager should prepare the required information for the Design Readiness Review.

4.3.3.4. Technical Reviews During System Integration

4.3.3.4.1. Integrated Baseline Review (IBR)

The program manager may convene an additional IBR to support the System Development and Demonstration contract. Section 4.3.2.4.2 of this Guidebook discusses the systems engineering considerations associated with an IBR. Section 11.3.4 describes an IBR in detail, and the Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics, in cooperation with industry, has prepared an IBR handbook.

4.3.3.4.2. System Requirements Review (SRR)

The SRR is a multi-functional technical review to ensure that all system and performance requirements derived from the Capability Development Document are defined and consistent with cost (program budget), schedule (program schedule), risk, and other system constraints. Generally this review assesses the system requirements captured in the system specification. The review ensures consistency between the system requirements and the preferred system solution and available technologies. The assigned manager may convene an SRR prior to program initiation, during Technology Development; and the program manager may convene an SRR during System Development and Demonstration. Section 4.3.2.4.1. of this Guidebook discusses the systems engineering considerations associated with an SRR.

4.3.3.4.3. System Functional Review (SFR)

The SFR is a multi-disciplined technical review to ensure that the system under review can proceed into preliminary design, and that all system requirements and functional performance requirements derived from the Capability Development Document are defined and are consistent with cost (program budget), schedule (program schedule), risk, and other system constraints. Generally this review assesses the system functional requirements as captured in system specifications (functional baseline), and ensures that all required system performance is fully decomposed and defined in the functional baseline. System performance may be decomposed and traced to lower-level subsystem functionality that may define hardware and software requirements. The SFR determines whether the systems functional definition is fully decomposed to a low level, and whether the IPT is prepared to start preliminary design.

Completion of the SFR should provide:

(1) An established system functional baseline;

(2) An updated risk assessment for the System Development and Demonstration phase;

(3) An updated Cost Analysis Requirements Description (CARD) (or CARD-like document) based on the system functional baseline;

(4) An updated program development schedule including system and software critical path drivers; and

(5) An approved Product Support Plan with updates applicable to this phase.

The SFR determines whether the system's lower-level performance requirements are fully defined and consistent with the mature system concept, and whether lower-level systems requirements trace to top-level system performance and the Capability Development Document. A successful SFR is predicated upon the IPT's determination that the system performance requirements, lower level performance requirements, and plans for design and development form a satisfactory basis for proceeding into preliminary design.

The program manager should tailor the review to the technical scope and risk of the system, and address the SFR in the Systems Engineering Plan. The SFR is the last review that ensures the system is credible and feasible before more technical design work commences.

Typical SFR success criteria include affirmative answers to the following exit questions:

(1) Can the system functional requirements, as disclosed, satisfy the Capability Development Document?

(2) Are the system functional requirements sufficiently detailed and understood to enable system design to proceed?

(3) Are adequate processes and metrics in place for the program to succeed?

(4) Are the risks known and manageable for development?

(5) Is the program schedule executable (technical/cost risks)?

(6) Is the program properly staffed?

(7) Is the program with the approved functional baseline executable within the existing budget?

(8) Is the updated Cost Analysis Requirements Description consistent with the approved functional baseline?

(9) Does the updated cost estimate fit within the existing budget?

(10) Has the system Functional Baseline been established to enable preliminary design to proceed with proper Configuration Management?

(11) Is the software functionality in the approved functional baseline consistent with the updated software metrics and resource loaded schedule?

4.3.3.4.4. Preliminary Design Review (PDR)

The PDR is a multi-disciplined technical review to ensure that the system under review can proceed into detailed design, and can meet the stated performance requirements within cost (program budget), schedule (program schedule), risk, and other system constraints. Generally, this review assesses the system preliminary design as captured in performance specifications for each configuration item in the system (allocated baseline), and ensures that each function in the functional baseline has been allocated to one or more system configuration items. Configuration items may consist of hardware and software elements and include such items as airframes, avionics, weapons, crew systems, engines, trainers/training, etc.

Completion of the PDR should provide:

(1) An established system allocated baseline;

(2) An updated risk assessment for System Development and Demonstration;

(3) An updated Cost Analysis Requirements Description (CARD) (or CARD-like document) based on the system allocated baseline;

(4) An updated program schedule including system and software critical path drivers; and

(5) An approved Product Support Plan with updates applicable to this phase.

For complex systems, the program manager may conduct a PDR for each subsystem or configuration item, leading to an overall system PDR. When individual reviews have been conducted, the emphasis of the overall system PDR should focus on configuration item functional and physical interface design, as well as overall system design requirements. The PDR determines whether the hardware, human, and software preliminary designs are complete, and whether the Integrated Product Team is prepared to start detailed design and test procedure development.

The PDR evaluates the set of subsystem requirements to determine whether they correctly and completely implement all system requirements allocated to the subsystem. The PDR also determines whether subsystem requirements trace with the system design. At this review the Integrated Product Team should review the results of peer reviews of requirements and preliminary design documentation. A successful review is predicated on the Integrated Product Team's determination that the subsystem requirements, subsystem preliminary design, results of peer reviews, and plans for development and testing form a satisfactory basis for proceeding into detailed design and test procedure development.

The program manager should tailor the review to the technical scope and risk of the system, and address the PDR in the Systems Engineering Plan.

Typical PDR success criteria include affirmative answers to the following exit questions:

(1) Does the status of the technical effort and design indicate operational test success (operationally suitable and effective)?

(2) Can the preliminary design, as disclosed, satisfy the Capability Development Document?

(3) Has the system allocated baseline been established and documented to enable detailed design to proceed with proper configuration management?

(4) Are adequate processes and metrics in place for the program to succeed?

(5) Have human integration design factors been reviewed and included, where needed, in the overall system design?

(6) Are the risks known and manageable for development testing and operational testing?

(7) Is the program schedule executable (technical/cost risks)?

(8) Is the program properly staffed?

(9) Is the program executable with the existing budget and with the approved system allocated baseline?

(10) Does the updated cost estimate fit within the existing budget?

(11) Is the preliminary design producible within the production budget?

(12) Is the updated Cost Analysis Requirements Description consistent with the approved allocated baseline?

(13) Is the software functionality in the approved allocated baseline consistent with the updated software metrics and resource-loaded schedule?

The program manager should conduct the PDR when all major design issues have been resolved and work can begin on detailed design. The PDR should address and resolved critical, system-wide issues.

4.3.3.4.5. Critical Design Review (CDR)

The CDR is a multi-disciplined technical review to ensure that the system under review can proceed into system fabrication, demonstration, and test; and can meet the stated performance requirements within cost (program budget), schedule (program schedule), risk, and other system constraints. Generally this review assesses the system final design as captured in product specifications for each configuration item in the system (product baseline), and ensures that each product in the product baseline has been captured in the detailed design documentation. Product specifications for hardware enable the fabrication of configuration items, and may include production drawings. Product specifications for software (e.g. Software Design Documents) enable coding of a Computer Software Configuration Item. Configuration items may consist of hardware and software elements, and include items such as airframe, avionics, weapons, crew systems, engines, trainers/training, etc. Completion of the CDR should provide:

(1) An established system product baseline;

(2) An updated risk assessment for System Development and Demonstration;

(3) An updated Cost Analysis Requirements Description (CARD) (or CARD-like document) based on the system product baseline;

(4) An updated program development schedule including fabrication, test, and software coding critical path drivers; and

(5) An approved Product Support Plan with updates applicable to this phase.

For complex systems, the program manager may conduct a CDR for each subsystem or configuration item. These individual reviews would lead to an overall system CDR. When individual reviews have been conducted, the emphasis of the overall system CDR should focus on configuration item functional and physical interface design, as well as overall system detail design requirements. The CDR determines whether the hardware, human, and software final detail designs are complete, and whether the Integrated Product Team is prepared to start system fabrication, demonstration, and test.

The subsystem detailed designs are evaluated to determine whether they correctly and completely implement all system requirements allocated to the subsystem, and whether the traceability of final subsystem requirements to final system detail design is maintained. At this review, the Integrated Product Team also reviews the results of peer reviews on requirements and final detail design documentation, and ensures that the latest estimates of cost (development, production, and support) are consistent with the detail design. A successful review is predicated on the Integrated Product Team's determination that the subsystem requirements, subsystem detail design, results of peer reviews, and plans for testing form a satisfactory basis for proceeding into system fabrication, demonstration and test.

The program manager should tailor the review to the technical scope and risk of the system, and address the CDR in the Systems Engineering Plan.

Typical CDR success criteria include affirmative answers to the following exit questions:

(1) Does the status of the technical effort and design indicate operational test success (operationally suitable and effective)?

(2) Does the detailed design, as disclosed, satisfy the CDD or any available draft CPD?

(3) Has the system product baseline been established and documented to enable hardware fabrication and software coding to proceed with proper configuration management?

(4) Has the detailed design satisfied Human Systems Integration (HSI) requirements?

(5) Are adequate processes and metrics in place for the program to succeed?

(6) Are the risks known and manageable for developmental testing and operational testing?

(7) Is the program schedule executable (technical/cost risks)?

(8) Is the program properly staffed?

(9) Is the program executable with the existing budget and the approved product baseline?

(10) Is the detailed design producible within the production budget?

(11) Is the updated CARD consistent with the approved product baseline?

(12) Are Critical Safety Items and Critical Application Items identified?

(13) Does the updated cost estimate fit within the existing budget?

(14) Is the software functionality in the approved product baseline consistent with the updated software metrics and resource-loaded schedule?

The program manager should conduct the CDR when the "build-to" baseline has been achieved, allowing production and coding of software deliverables to proceed.

4.3.3.5. Outputs of the Systems Engineering Processes/Inputs to the Design Readiness Review

The outputs of the systems engineering processes in System Integration become the inputs to the Design Readiness Review. These inputs include the following measures of design maturity:

- The number of subsystem and system technical reviews successfully completed;
- The percentage of drawings completed;
- Planned corrective actions to hardware/software deficiencies;
- Adequate development testing;
- An assessment of environment, safety and occupational health risks;
- A completed failure modes and effects analysis;
- The identification of key system characteristics and critical manufacturing processes; and
- An estimate of system reliability based on demonstrated reliability rates; etc.

4.3.3.6. Purpose of Systems Engineering in System Demonstration

Successful completion of the Design Readiness Review and successful demonstration of the system in prototypes or engineering development models end System Integration work effort. The program will normally continue in the System Development and Demonstration phase with the System Demonstration effort. System Demonstration demonstrates the ability of the system to operate in a useful way consistent with the approved key performance parameters. Through the use of systems engineering, a system is demonstrated in its intended environment, using the selected prototype. When the necessary industrial capabilities are reasonably available, the system satisfies approved requirements, and the system meets or exceeds exit criteria and Milestone C entrance requirements, the System Demonstration effort may end. Key to the System Demonstration effort is acceptable performance in developmental test and evaluation and early operational assessments, and the use of modeling and simulation to support test design and the demonstration of satisfactory system integration.

4.3.3.7. Inputs to the Systems Engineering Processes in System Demonstration

The results of the Design Readiness Review provide the principal inputs to the systems engineering processes during System Demonstration. The Capability Production Document, finalized after the Design Readiness Review, provides additional input.

4.3.3.8. Key SE Activities During System Demonstration

Figure 18 illustrates the steps during the System Demonstration part of the System Development and Demonstration phase. Further detail on each step is contained in paragraphs 4.3.3.8.1. through 4.3.3.8.4.

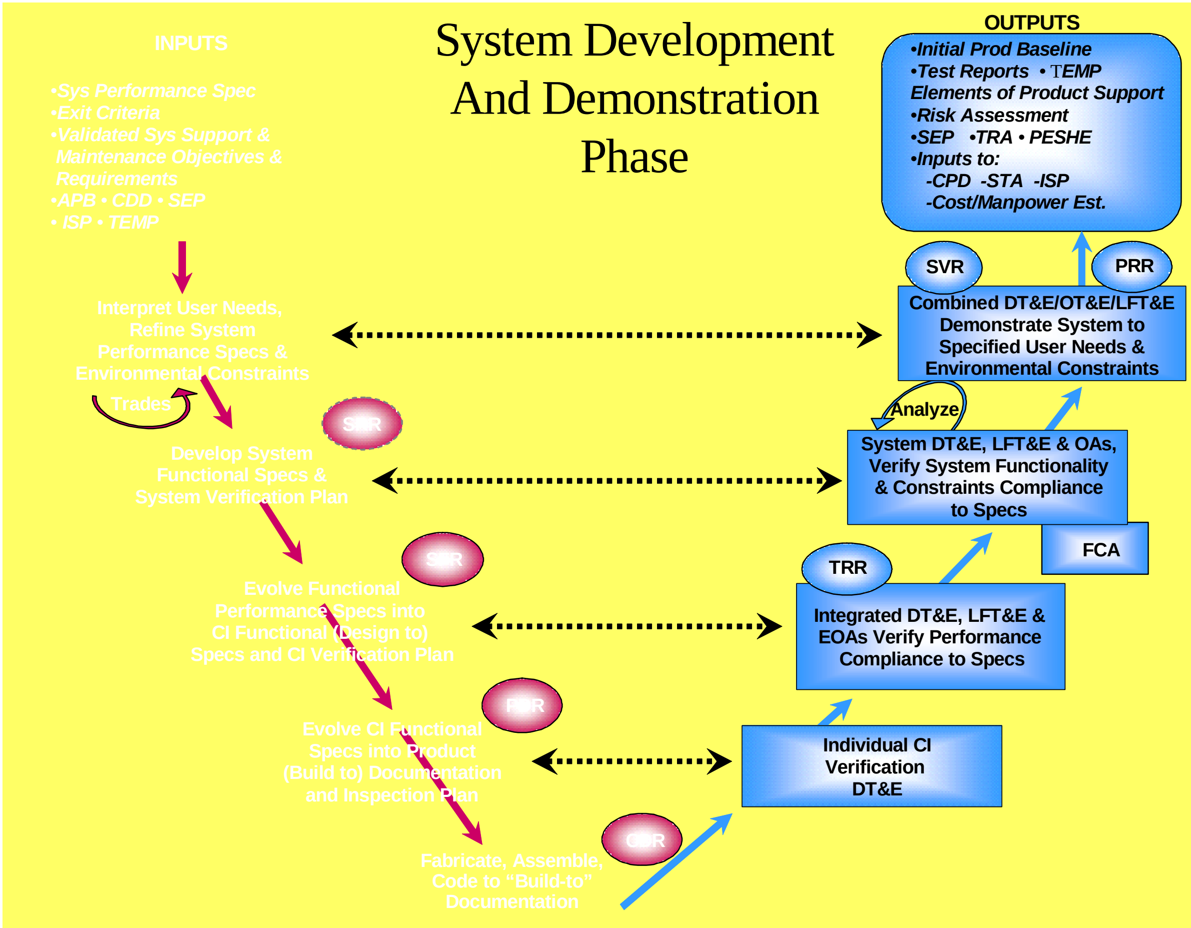

Figure 4. Systems engineering-related steps during the System Demonstration effort of System Development and Demonstration.

4.3.3.8.1. Developmental Test and Evaluation verifies Individual Configuration Items

Demonstrate, according to the verification and validation plans, the physical, electrical, software, and other characteristics of the components to be integrated. Begin unit testing of hardware and independent verification and validation of software. Special attention should be placed on the integration and testing of commercial components. Ensure the components and any assemblies of them meet their requirements and function in the environment of their intended use. Developmental test and evaluation is conducted on the configuration items to assess technical progress against critical technical parameters. Continue to monitor risk, cost, and schedule. Design issues that arise as a result of the Integration, Verification, or Validation processes should feed back into the Design Solution process for refinement to the design. Early component level test may not require the same level of review as the final system level tests.

4.3.3.8.2. Integrated Developmental Test and Evaluation, Live Fire Test and Evaluation, and Early Operational Assessments verify Performance Compliance to Specifications

Verify subsystem hardware and software performance against their defined subsystem design requirements. Demonstrate subsystem hardware and software in their intended environment. Early operational assessments and developmental test and evaluation are conducted at the subsystem level, and risk, cost, and schedule continue to be monitored.

The Test Readiness Review occurs after this activity. The program manager determines the "formality" and scope of the Test Readiness Review for each assembly or subsystem.

The program manager also conducts the Functional Configuration Audit to verify that the actual performance of the configuration item meets specification requirements.

4.3.3.8.3. System Developmental Test and Evaluation, Live Fire Test and Evaluation, and Operational Assessments verify System Functionality and Constraints Compliance to Specifications

Integrate the subsystems into the defined system and demonstrate the integrated system under its operational environment constraints. This verifies that the system meets performance and functionality requirements, and validates the use of the system in its intended environment. This step includes developmental test and evaluation, any live fire test and evaluation, and operational assessments on the integrated system. All integration and interface issues must be resolved. Monitor and analyze risks as they pertain to the cost, schedule, and performance of the integrated system.

4.3.3.8.4. Combined Developmental Test and Evaluation, Operational Test and Evaluation, and Live Fire Test and Evaluation Demonstrate System to Specified User Needs and Environmental Constraints

Verify and validate the integrated system against the specified operational requirements within the required operational environment(s) to ensure the system can satisfy operational expectations. The developmental and operational test environments and scenarios must be defined, and cost, schedule, and performance considerations must be continually addressed. This involves interoperability and interfaces for the system within any system of systems in which it operates. Any interface and interoperability issues for the system must be resolved for the system to achieve its interoperability certification in the next phase. Operational supportability should be confirmed at this time. In preparation for the Production Readiness Review, this step should confirm that the manufacturing processes are under control and that there are no significant manufacturing risks. Technical risk must be addressed, characterized, and mitigated.

4.3.3.9. Technical Reviews During System Demonstration

4.3.3.9.1. Test Readiness Review (TRR)

The TRR is a multi-disciplined technical review to ensure that the subsystem or system under review is ready to proceed into formal test. The TRR assesses test objectives, test methods and procedures, scope of tests, and determines if required test resources have been properly identified and coordinated to support planned tests. The TRR verifies the traceability of planned tests to program requirements and user needs. The TRR determines the completeness of test procedures and their compliance with test plans and descriptions. The TRR assesses the system under review for development maturity, cost/ schedule effectiveness, and risk to determine readiness to proceed to formal testing. In addition to adequate planning and management, to be effective the program manager should follow-up with the outcomes of the TRR.

Test and evaluation is an integral part of the systems engineering processes of Verification and Validation. Test and evaluation should permeate the entire life cycle of an acquisition program.

Test and evaluation is also an important tool to identify and control risk.

This discussion principally addresses the TRR to support the readiness for a system to proceed into system-level Developmental Test. However, the program manager could utilize the TRR process to support all tests in all phases of an acquisition program, including testing within a system of systems context. A robust test program should enhance the program manager's ability to identify and manage risk. The program managers and Test and Evaluation Working-level Integrated Product Team should tailor any TRR to the specific acquisition phase, the specific planned tests, and the identified level of risk within the program. The scope of the review is directly related to the risk level associated with performing the planned tests and the importance of the test results to overall program success. The program manager should address the scope of the TRR(s) in the Systems Engineering Plan.

The level of specific risk and risk level will vary as a system proceeds from component level, to system level, to systems of systems level testing. Early component level test may not require the same level of review as the final system level tests. Sound judgment should dictate the scope of a specific test or series of tests.

Readiness to convene a TRR is predicated on the program manager’s and Test and Evaluation Working-level Integrated Product Team’s determination that preliminary testing, functional testing, and pre-qualification testing results form a satisfactory basis for proceeding with a TRR and subsequent initiation of formal, system-level Developmental Test.

As a practical matter, the program manager should carefully plan and properly resource test events.

Regardless of stage of development or the level of the testing (component, subsystem, or system), the basic tenets of this discussion about the TRR should apply.

The TRR should answer the following questions:

(1) Why are we testing? What is the purpose of the planned test? Does the planned test verify a requirement that is directly traceable back to a system specification or other program requirement?

(2) What are we testing (subsystem, system, system of systems, other)? Is the configuration of the system under test sufficiently mature, defined, and representative to accomplish planned test objectives and or support defined program objectives?

(3) Are we ready to begin testing? Have all planned preliminary, informal, functional, unit level, subsystem, system, and qualification tests been conducted, and are the results satisfactory?

(4) What is the expected result and how can/do the test results affect the program?

(5) Is the planned test properly resourced (people, test article or articles, facilities, data systems, support equipment, logistics, etc.)

(6) What are the risks associated with the tests and how are they being mitigated?

(7) What is the fall-back plan should a technical issue or potential showstopper arise during testing?

Typical TRR success criteria include:

(1) Completed and approved test plans for the system under test;

(2) Completed identification and coordination of required test resources;

(3) The judgment that previous component, subsystem, and system test results form a satisfactory basis for proceeding into planned tests; and

(4) Identified risk level acceptable to the program leadership.

Test and evaluation is critical to evaluating the system. The TRR ensures that the testing to be conducted properly evaluates the system and that the system is ready to be tested.

4.3.3.9.2. System Verification Review (SVR)

The SVR (synonymous with Functional Configuration Audit) is a multi-disciplined technical review to ensure that the system under review can proceed into Low-Rate Initial Production and Full-Rate Production within cost (program budget), schedule (program schedule), risk, and other system constraints. Generally this review is an audit trail from the Critical Design Review. It assesses the system final product, as evidenced in its production configuration, and determines if it meets the functional requirements (derived from the Capability Development Document and draft Capability Production Document) documented in the Functional, Allocated, and Product Baselines. The SVR establishes and verifies final product performance. It provides inputs to the Capability Production Document. The SVR is often conducted concurrently with the Production Readiness Review.

Typical SVR success criteria include affirmative answers to the following exit questions:

(1) Does the status of the technical effort and system indicate operational test success (operationally suitable and effective)?

(2) Can the system, as it exists, satisfy the Capability Development Document/draft Capability Production Document?

(3) Are adequate processes and metrics in place for the program to succeed?

(4) Are the risks known and manageable?

(5) Is the program schedule executable within the anticipated cost and technical risks?

(6) Are the system requirements understood to the level appropriate for this review?

(7) Is the program properly staffed?

(8) Is the program's Non Recurring Engineering requirement executable with the existing budget?

(9) Is the system producible within the production budget?

4.3.3.9.3. Production Readiness Review (PRR)

The PRR examines a program to determine if the design is ready for production and if the producer has accomplished adequate production planning. The review examines risk; it determines if production or production preparations incur unacceptable risks that might breach thresholds of schedule, performance, cost, or other established criteria. The review evaluates the full, production-configured system to determine if it correctly and completely implements all system requirements. The review determines whether the traceability of final system requirements to the final production system is maintained.

At this review, the Integrated Product Team should review the readiness of the manufacturing processes, the Quality Management System, and the production planning (i.e. facilities, tooling and test equipment capacity, personnel development and certification, process documentation, inventory management, supplier management, etc.) A successful review is predicated on the Integrated Product Team's determination that the system requirements are fully met in the final production configuration, and that production capability forms a satisfactory basis for proceeding into Low-Rate Initial Production and Full-Rate Production.

The program manager should convene a PRR of the prime contractor *and* major subcontractors, as applicable. The PRR(s) should be conducted in an iterative fashion, concurrently with other technical reviews, such as the System Functional Review, the Preliminary Design Review, and the Critical Design Review, during the System Development and Demonstration phase. Periodic production readiness assessments should be conducted during the System Demonstration work effort to identify and mitigate risks as the design progresses. The "*final*" PRR should occur at the completion of the System Development and Demonstration phase and the start of the Production and Deployment Phase. The final PRR should assess the manufacturing and quality risk as the program proceeds into Low-Rate Initial Production and Full-Rate Production.

The program manager should tailor the PRR to the technical scope and risk associated with the system. The program manager should address the PRR in the Systems Engineering Plan.

Typical PRR success criteria include affirmative answers to the following exit questions:

(1) Has the system product baseline been established and documented to enable hardware fabrication and software coding to proceed with proper configuration management?

(2) Are adequate processes and metrics in place for the program to succeed?

(3) Are the risks known and manageable?

(4) Is the program schedule executable (technical/cost risks)?

(5) Is the program properly staffed?

(6) Is the detailed design producible within the production budget?

A follow-on, tailored, PRR may be appropriate in the Production and Deployment phase for the prime contractor and major subcontractors if:

(1) Changes from the System Development and Demonstration phase and during the production stage of the design, in either materials or manufacturing processes, occur;

(2) Production start-up or re-start occurs after a significant shutdown period;

(3) Production start-up with a new contractor; or

(4) Relocation of a manufacturing site.

4.3.3.9.4. Technology Readiness Assessment (TRA)

The program manager should normally conduct a second TRA prior to Milestone C. The TRA may be held concurrently with other technical reviews, specifically System Requirements Review, Critical Design Review, System Verification Review, or Production Readiness Review. Completion of this TRA should provide:

(1) An evaluation of system technology maturity based on the Work Breakdown Structure;

(2) An objective scoring of the level of technological maturity; and

(3) Mitigation plans for achieving acceptable maturity prior to milestone decision dates.

4.3.3.10. Outputs of the Systems Engineering Processes in System Development and Demonstration

- Initial Product Baseline;
- Test Reports;
- Test and Evaluation Master Plan;
- Elements of Product Support (add this link here: http://acc.dau.mil/simplify/file_download.php/FINAL+GUIDE+with+Memo+-+October+24.pdf?URL_ID=15943&filename=10772113271FINAL_GUIDE_with_Memo_-_October_24.pdf&filetype=application%2Fpdf&filesize=432407&name=FINAL+GUIDE+with+Memo+-+October+24.pdf&location=user-S/#page=22;
- Risk Assessment;
- Systems Engineering Plan;
- Technology Readiness Assessment;
- Programmatic Environment, Safety, and Occupational Health Evaluation;
- Inputs to the Capability Production Document;
- Inputs to System Threat Assessment;
- Inputs to the Information Support Plan; and
- Inputs to Cost and Manpower Estimate.

4.3.4. Production and Deployment Phase

The Production and Deployment Phase commences at Milestone C and ends with the declaration of Initial Operational Capability. During the Production and Deployment Phase, the system should achieve operational capability that satisfies mission needs.

Two work efforts, separated by the Full-Rate Production Decision Review, comprise the Production and Deployment Phase: Low-Rate Initial Production and Full-Rate Production and Deployment.

4.3.4.1. Purpose of Systems Engineering in Production and Deployment

As the integrated components develop into a system, the test and evaluation processes frequently reveal issues that require improvements or redesign. As the testing environment more closely approaches that of the users needs, the required improvements might be complex and/or subtle. The initial manufacturing process may also reveal issues that were not anticipated. It may be discovered that changing the product somewhat may provide enhancements in the manufacturing or other supporting processes. Low-Rate Initial Production should result in completion of manufacturing development. The systems engineering effort in Full-Rate Production and Deployment delivers the fully-funded quantity of systems and supporting materiel and services for the program or increment. During this effort, units attain Initial Operational Capability.

4.3.4.2. Inputs to the Systems Engineering Processes in Production and Deployment

- Test Results
- Exit Criteria to leave the Production and Deployment phase and enter the Operations and Support phase
- Acquisition Program Baseline
- Capability Development Document and Capability Production Document
- Systems Engineering Plan
- Test and Evaluation Master Plan
- Product Support Package (add this link here: http://acc.dau.mil/simplify/file_download.php/FINAL+GUIDE+with+Memo+-+October+24.pdf?URL_ID=15943&filename=10772113271FINAL_GUIDE_with_Memo_-_October_24.pdf&filetype=application%2Fpdf&filesize=432407&name=FINAL+GUIDE+with+Memo+-+October+24.pdf&location=user-S/#page=22;

4.3.4.3. Key Systems Engineering Activities During Production and Deployment

Figure 19 illustrates the steps during the Production and Deployment phase. Some activities and reports are shown outside of the systems engineering V-shaped model that was used in describing the other phases. The following paragraphs, 4.3.4.3.1. through 4.3.4.3.3, contain further detail on each step. The Test Readiness Review and Physical Configuration Audit are covered in Sections 4.3.3.9.1 and 4.3.4.4.3, respectively.

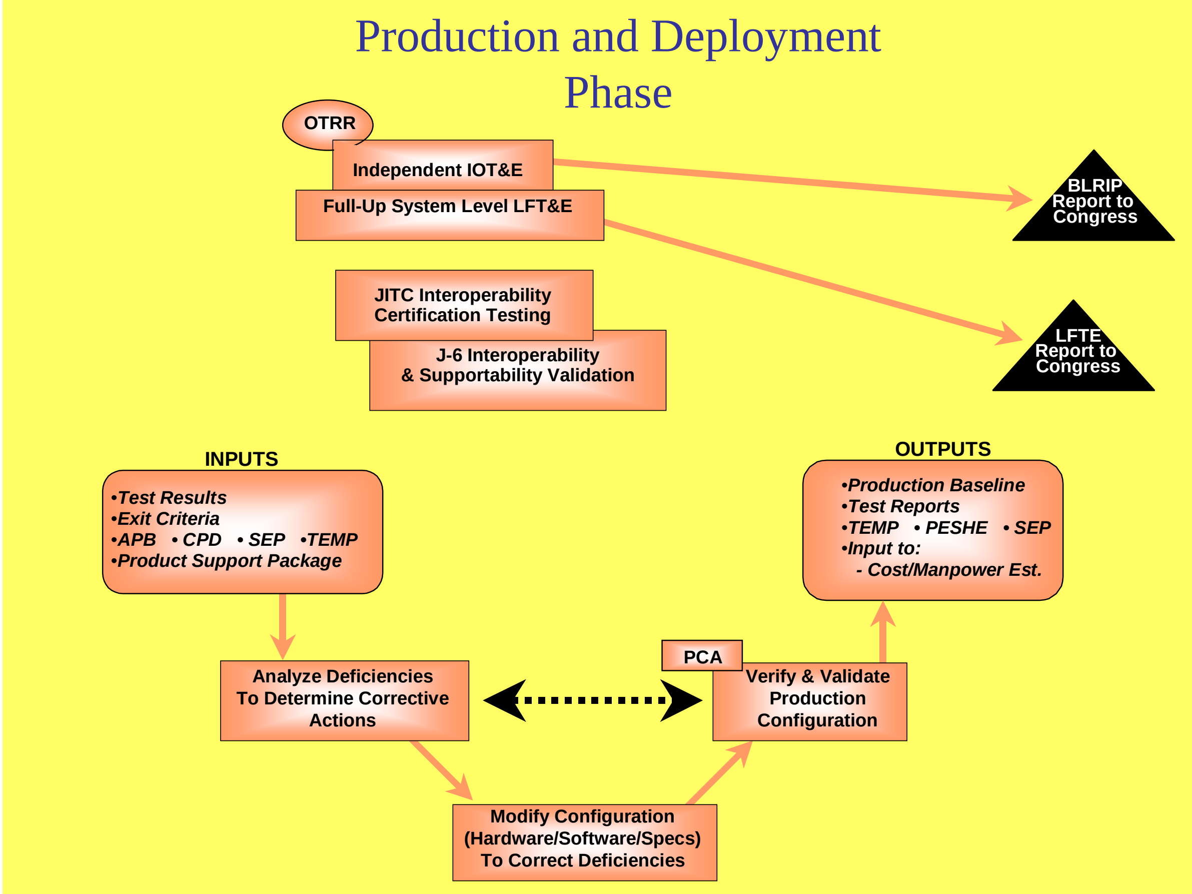

Figure 5. Systems Engineering Activities During Production and Deployment.

4.3.4.3.1. Analyze Deficiencies to Determine Corrective Actions

Using the aggregation of all inputs available at this stage of the program (test results, maintenance reports, exit criteria from System Development and Demonstration, Capability Production Document, Systems Engineering Plan, Test and Evaluation Master Plan, as well as associated support and maintenance concepts), known deficiencies are analyzed. A solution is proposed through the employment of Systems Engineering processes. A plan to build/modify/verify, and test the proposed solution is formulated and approved.

4.3.4.3.2. Modify Configuration (Hardware, Software, and Specifications) to Correct Deficiencies

The proposed solution to the deficiency is translated to the appropriate hardware/software or specification changes. Modifications are created, incorporated, and verified in accordance with the approved plan. This product change may include retrofit, since the production process has begun. The impact on system cost, schedules, and performance should also be considered when addressing production incorporation.

4.3.4.3.3. Verify and Validate Production Configuration

The proposed solution to the system deficiency should be verified and tested. This process may require the spectrum from laboratory through full operational system testing. These test, analyze and fix activities may have to be repeated to resolve deficiencies or further improve the system solution. These approved changes should be incorporated into the final production configuration baseline.

4.3.4.4. Technical Reviews During Production an Deployment

4.3.4.4.1. Integrated Baseline Review (IBR)

The program manager may convene an additional IBR to support the Low-Rate Initial Production contract. Section 4.3.2.4.2. of this Guidebook discusses the systems engineering considerations associated with an IBR. Section 11.3.4. describes an IBR in detail. The Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics, in cooperation with industry has also prepared an IBR handbook.

Completion of IBR at this stage of the life cycle should result in the assessment of risk and the degree to which the six criteria described in 4.3.2.4.2 are met.

4.3.4.4.2. Operational Test Readiness Review (OTRR)

The program manager may conduct another TRR around the time of the Milestone C decision. The OTRR is a multi-disciplined product and process assessment to ensure that the "production configuration" system can proceed into Operational Test and Evaluation (OT&E) with a high probability of successfully completing the operational testing. Successful performance during operational test generally indicates that the system is suitable and effective for service introduction. The Full Rate Production Decision may hinge on this successful determination. The understanding of available system performance to meet the CPD is important to the OTRR. The OTRR is complete when the Service Acquisition Executive evaluates and determines materiel system readiness for IOT&E.

4.3.4.4.3. Physical Configuration Audit (PCA)

The PCA is conducted around the time of the full rate production decision. The PCA examines the actual configuration of an item being produced. It verifies that the related design documentation matches the item as specified in the contract. In addition to the standard practice of assuring product verification, the PCA confirms that the manufacturing processes, quality control system, measurement and test equipment, and training are adequately planned, tracked, and controlled. The PCA validates many of the supporting processes used by the contractor in the production of the item and verifies other elements of the item that may have been impacted/redesigned after completion of the System Verification Review (SVR). A PCA is normally conducted when the government plans to control the detail design of the item it is acquiring via the Technical Data Package. When the government does not plan to exercise such control or purchase the item's Technical Data Package (e.g., performance based procurement) the contractor should conduct an internal PCA to define the starting point for controlling the detail design of the item and establishing a product baseline. The PCA is complete when the design and manufacturing documentation match the item as specified in the contract. If the PCA was not conducted prior to the full rate production decision, it should be performed as soon as production systems are available.

4.3.4.5. Outputs of the Systems Engineering Processes in Production and Deployment

- Production Baseline;
- Test Reports;
- Test and Evaluation Master Plan;

- Programmatic Environment, Safety, and Occupational Health Evaluation;
- Systems Engineering Plan; and
- Inputs to Cost and Manpower Estimate.

4.3.5. Operations and Support Phase

The objective of this phase is the execution of a support program that meets operational support performance requirements and sustains the system in the most cost-effective manner over its total life cycle. When the system reaches the end of its useful life, the Department must dispose of it. These two work efforts, Sustainment and Disposal, comprise the Operations and Support Phase.

4.3.5.1. Purpose of Systems Engineering in Operations and Support

During the Sustainment effort of the Operations and Support Phase, systems engineering processes support in-service reviews, trade studies, and decision making on modifications, upgrades, and future increments of the system. Interoperability or technology improvements, parts or manufacturing obsolescence, aging aircraft (or system) issues, premature failures, changes in fuel or lubricants, Joint or service commonality, etc. may all indicate the need for a system upgrade(s).

System disposal is not a systems engineering activity. However, systems engineering processes that inject disposal requirements and considerations into the earlier design processes ultimately address and impact disposal.

4.3.5.2. Inputs to the Systems Engineering Processes in Operations and Support

- Service Use Data;
- User feedback;
- Failure reports;
- Discrepancy reports; and
- Systems Engineering Plan.

4.3.5.3. Key Systems Engineering Activities During Operations and Support

Figure 20 illustrates the steps during the Operations and Support phase. Further detail on each step is contained in paragraphs 4.3.5.3.1. through 4.3.5.3.7. Systems engineering should continue during operation and support of the system, and be used to continuously assess fielded system technical health against documented performance requirements and effectiveness, suitability, and risk measures. In-service systems engineering provides the program manager with an integrated technical assessment of system trends and sustainment alternatives, and then is used to oversee development and implementation of the selected alternative.

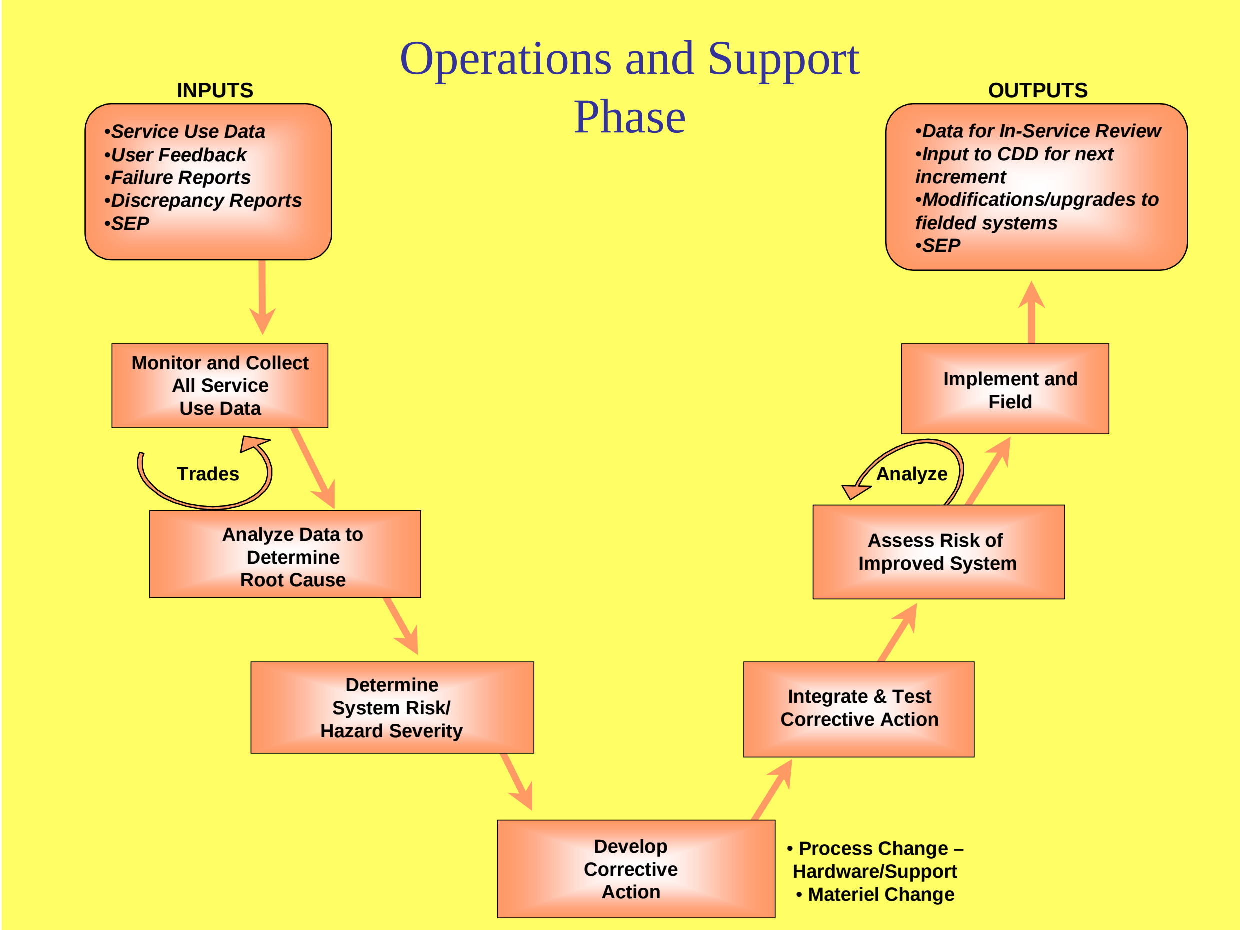

Figure 6. Systems Engineering Activities During Operations and Support.

4.3.5.3.1. Monitor and Collect All Service Use Data

The aggregation of all data inputs available at this stage of the program (service use data, maintenance discrepancy reports, user feedback, system/component failure reports, and the Systems Engineering Plan) provides the life cycle basis for many O&S decisions that will be made throughout the operational life of the system. Historically, many fielded systems remain in service much longer than originally planned. The type of data retrieved may change as the operational understanding of the system matures.

4.3.5.3.2. Analyze Data to Determine Root Cause of Problem

As problems arise in the fielded system, the systems engineering processes determine the cause of the problem and may lead to a solution. The retrieved data is key to this determination, and should be thoroughly analyzed for causes and potential solutions. These analyses may ascertain whether deficiencies exist in the system as designed/built, or whether the system has been operated differently, or in a different environment, than that for which it was designed.

4.3.5.3.3. Determine the System Risk/Hazard Severity

Risk assessment techniques and principles, as well as systems engineering processes determine the hardware/software safety hazards, and identify the readiness, program, and cost risks associated with the identified problems and/or deficiencies.

4.3.5.3.4. Develop Corrective Action

Corrective actions may include process, hardware, software, support, materiel, or maintenance changes. The systems engineering process is utilized to develop appropriate corrective actions.

4.3.5.3.5. Integrate and Test Corrective Action

Integrate the proposed corrective process, hardware, software, support, materiel, and/or maintenance changes; and methodically test the resultant prototype. Adequate testing (regression, durability, functional, interoperability, etc.) should be completed to ensure the proposed corrective action is suitable for fielding.

4.3.5.3.6. Assess Risk of Improved System

Once the functionality of the proposed corrective action is demonstrated, long-range system ramifications should be addressed. The appropriate systems engineering process is a risk assessment, which involves in-depth (regression, durability, structural, interoperability, support, etc.) system analyses. Additionally, the support, training, documentation, configuration control, and maintenance aspects of the improvements should be considered. All of these elements have an impact on system life cycle costs, which should be meticulously calculated in order to justify the required funding.

4.3.5.3.7. Implement and Field

The system corrective action/improvement may be authorized, implemented, and fielded once the correction/improvement is thoroughly understood and tested, and adequate supplies, support, training, and maintenance procedures are provided. Documentation and configuration control should be thorough and meticulous. This data is utilized during periodic In-Service Reviews (ISRs) to document in-service health, operational system risk, system readiness, costs, trends, aging equipment and out of production issues.

4.3.5.4. Technical Reviews During Operations and Support

4.3.5.4.1. In-Service Review (ISR)

The ISR is a multi-disciplined product and process assessment to ensure that the system under review is operationally employed with well-understood and managed risk. This review is intended to characterize the in-service technical and operational health of the deployed system. It provides an assessment of risk, readiness, technical status, and trends in a measurable form. These assessments substantiate in-service support budget priorities. The consistent application of sound programmatic, systems engineering, and logistics management plans, processes, and sub-tier in-service stakeholder reviews will help achieve the ISR objectives. Example support groups include the System Safety Working Group and the Integrated Logistics Management Team. A good supporting method is the effective use of available government and commercial data sources. In-service safety and readiness issues are grouped by priority to form an integrated picture of in-service health, operational system risk, system readiness, and future in-service support requirements.

The ISR should provide:

(1) An overall System Hazard Risk Assessment;

(2) An operational readiness assessment in terms of system problems (hardware, software, and production discrepancies); and

(3) Status of current system problem (discrepancy) report inflow, resolution rate, trends, and updated metrics. The metrics may be used to prioritize budget requirements.

Successful completion of this review should provide the Program Manager and other stakeholders with the integrated information they need to establish priorities and to develop execution and out year budget requirements.

Typical success outcomes include:

(1) System problems have been categorized to support the O&S requirements determination process.

(2) Required budgets (in terms of work years) have been established to address all system problems in all priority categories.

(3) Current levels of System Operational Risk and System Readiness have been quantified and related to current O&S and procurement budgets.

(4) Future levels of System Operational Risk and System Readiness have been quantified and related to future year O&S and procurement budgets.

4.3.5.5. Outputs of the SE Processes in Operations and Support

- Input to Capability Development Document for next increment of the system;
- Modifications and upgrades to fielded systems; and
- Systems Engineering Plan.

4.3.6. Evolutionary Acquisition Programs

Programs with an evolutionary acquisition strategy undergo additional reviews (e.g., a MS B decision for each increment). The systems engineering activities and reviews are repeated as appropriate to ensure the same level of program insight is achieved within Evolutionary Acquisition Programs.

4.4. Systems Engineering Decisions: Important Design Considerations

The program manager faces a myriad of considerations and management tools to translate the user's desired capabilities (regardless of phase in the acquisition cycle) into a structured system of interrelated design specifications. This is clearly not a trivial task. It is an iterative task, performed within the framework of Systems Engineering to achieve the "best value" for the user.

The "best value" solution is not an easy solution to define. Many requirements and design considerations cannot fully coexist in a single design – hence, the need for rigorous systems engineering processes with trade offs. The systems engineering processes detailed in Section 4.2 and applied in each acquisition phase as detailed in Section 4.3 will enable the program manager to manage expectations of the user across the spectrum of requirements and design. The systems engineering management tools discussed in Section 4.5 give the program manager the methodology to examine the specific characteristics of his/her own program against a myriad of often-conflicting design considerations. This section discusses a number of these considerations and how they contribute to program performance. Each will have a different, "optimal" solution depending on the capabilities required of the program. Some "design considerations" will take the form of design constraints (e.g., weight, volume, power, cooling, etc.) that are derived requirements and need to be closely managed through a rigorous trades process. Some constraints may form system-wide budgets and require close tracking as the design matures. The challenge for the program manager is to apply systems engineering to achieve balance across all of the considerations and constraints.

The program manager should be aware that some considerations are mandated by law and others will be mandated by the user in the program's capability document. *These mandates must be preeminent in the program manager's design considerations balancing act.*

Figure 7 provides a framework for how these design considerations fit into an affordable systems operational effectiveness framework.

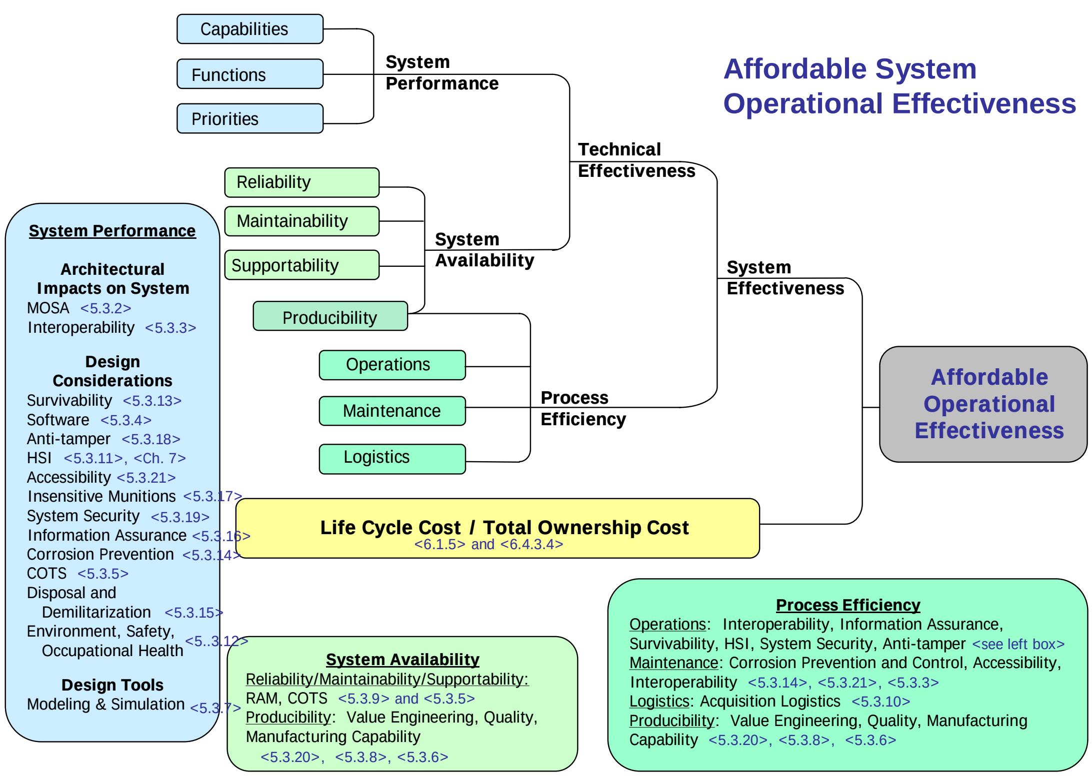

Figure 7. Affordable System Operational Effectiveness Diagram

4.4.1. Open Systems Design

An open system is a system that implements specifications maintained by an open, public consensus process for interfaces, services, and support formats, to enable properly engineered components to be utilized across a wide range of systems with minimal changes. The open systems approach should be an integral part of the overall acquisition strategy to enable rapid acquisition with demonstrated technology, evolutionary and conventional development, interoperability, life-cycle supportability, and incremental system upgradeability without major redesign during initial procurement and reprocurement of systems, subsystems, components, spares, and services during post-production support.

An open systems design is a means to assess and implement, when feasible, widely supported commercial interface standards in developing agile architectures for systems using modular design tenets. Program managers should employ an open systems design strategy within the context of implementing their overall Modular, Open Systems Approach. Within the context of MOSA, a program would employ an open systems design strategy only after careful analysis of required capabilities, economic justifications, and market research findings point to its appropriateness. The program manager should ensure that capabilities required lend themselves to the application of open systems design and do not impose premature design specific solutions. The program manager also needs to make a business case for using the open systems design through the application of dynamic cost models and conduct of market research aimed at analyzing the availability of open standards and the degree of market support for such standards.

Successful implementation of an open systems design strategy depends on planning for MOSA. Such planning should consider (1) how MOSA fits into the program's overall acquisition process and strategies for acquisition, technology development, and T&E; (2) what steps the program will take to analyze, develop, and implement an architecture based on MOSA principles; and (3) how the program intends to assess its MOSA implementation progress and assure openness of its system's architecture. Programs should develop the architecture for their systems based on adherence to the following four MOSA principles:

- Employ modular design. Determine the degree to which modules are cohesive (contain well-focused and well-defined functionality); encapsulated (hide the internal workings of a module's behavior and its data); self-contained (do not constrain other modules); and highly binded (use broad modular definitions to enable commonality and reuse). By following these tenets, each module will be designed for change and the interface to each module is defined in such a way as to reveal as little as possible about its inner workings which facilitate the standardization of modular interfaces.
- Designate key interfaces. To effectively manage hundreds and in some cases thousands of interfaces that exist within and among systems, designers should group interfaces into key and non-key interfaces. Such distinction enables designers and configuration managers to distinguish among interfaces that exist between technologically stable and volatile modules, between highly reliable and more frequently failing modules, between modules that are essential for net-centricity and those that do not perform net-centric functions, and between modules that pass vital interoperability information and those with least interoperability impact.

- Use open standards. In order to take full advantage of modularity in design, interface standards must be well defined, mature, widely used, and readily available. Moreover, standards should be selected based on maturity, market acceptance, and allowance for future technology insertion. As a general rule, preference is given to the use of open interface standards first, the de facto interface standards second, and finally government and proprietary interface standards. Basing design strategies on widely supported open standards increases the chance that future changes will be able to be integrated in a cost effective manner.
- Ensure openness. Openness of systems is verified, validated, and ensured through rigorous and well-established assessment mechanisms, well-defined interface control and management, and proactive conformance and confirmation testing. Open systems verification and validation should become an integral part of the overall organization change and configuration management processes. Program managers should preferably use the MOSA PART developed by the Open Systems Joint Task Force to assess the compliance with open systems policies and ensure that their programs are properly positioned to reap the open systems benefits.

These principles are recommended to ensure access to the latest technologies and products, achieve interoperability, and facilitate affordable and supportable modernization of fielded assets. They also help ensure delivery of technologically superior, sustainable, and affordable increments of militarily useful capability within an evolutionary acquisition strategy context. For more information and detailed guidance on using MOSA and open systems design, please see the Open Systems Joint Task Force guide.

4.4.2. Interoperability

All acquisition programs are required to satisfactorily address interoperability and integration. These requirements span the complete acquisition life cycle for all acquisition programs. Interoperability and supportability of information technology (IT) and National Security System (NSS) acquisition programs, are required to comply with DoD Directive 4630.5, DoD Instruction 4630.8, CJCS Instruction 3170.01C, CJCS Manual 3170.01 , CJCSI 6212.01C), Public Law 104-106 (1996), and 44 U.S.C. 3506.

4.4.3. Standardization

Standardization advances interoperability through commonality of systems, subsystems, components, equipment, data, and architectures. The program manager balances decisions to use standard systems, subsystems, and support equipment against specific capabilities (including corresponding information system elements that perform critical essential, or support functions within each joint functional capability), technology growth, and cost effectiveness.

Program managers should consider compliance with international standardization agreements, such as the NATO Standardization Agreements, or the agreements of the Air Standards Coordinating Committee or American-British-Canadian-Australian Armies. The program manager should identify any international standardization agreements or U.S. implementing documents that apply to the program early in the design process to ensure interoperability with combined and coalition systems and equipment. The program manager should employ systems engineering analysis if compliance with the DoD Joint Technical Architecture or other international standardization agreements and/or other standards does not provide sufficient interoperability to satisfy user requirements.

4.4.4. Software

The program manager should base software systems development on robust systems engineering principles. The following best practices for software systems also apply in general to any system:

- Viewing the software "content," particularly complex algorithms and functional flows, as enabling technologies requiring maturation and risk reduction prior to MS B;
- Developing architectural-based software systems that support open system concepts;
- Exploiting commercial, off-the-shelf (COTS) computer systems products;
- Allowing incremental improvements based on modular, reusable, extensible software;
- Identifying and exploiting, where practicable, Government and commercial software reuse opportunities before developing new software;
- Selecting the programming language in context of the systems and software engineering factors that influence overall life-cycle costs, risks, and the potential for interoperability;
- Using DoD standard data and following data administrative policies in DoD Directive 8320.1;
- Selecting contractors with domain experience in developing comparable software systems; with successful past performance; and with a mature software development capability and process;
- Assessing information operations risks (see DoD Directive S-3600.1) using techniques such as independent expert reviews;
- Preparing for life-cycle software support or maintenance by developing or acquiring the necessary documentation, host systems, test beds, and computer-aided software engineering tools consistent with planned support concepts;
- Preparing for life-cycle software support or maintenance by planning for transition of fielded software to the support/maintenance activity; and
- Tracking COTS software purchases and maintenance licenses.

The program manager should structure a software development process to recognize that emerging capabilities and missions will require modification to software over the life cycle of the system. In order to deliver truly state-of-the-software, this process should allow for periodic software enhancements.

Additionally, the program manager should apply the following security considerations to software design and management (see DoD Directive 5000.1):

- A documented impact analysis statement, which addresses software reliability and accompanies modifications to existing DoD software;
- Formal software change control processes;

  - Software quality assurance personnel monitor the software change process;
  - An independent verification and validation team provides additional review;
- A change control process indicating whether foreign nationals, in any way, participated in software development, modification, or remediation;
- Each foreign national employed by contractors/subcontractors to develop, modify, or remediate software code specifically for DoD use has a security clearance commensurate with the level of the program in which the software is being used;
- Primary vendors on DoD contracts that have subcontractors who employ cleared foreign nationals work only in a certified or accredited environment (DoD Instruction 5200.40);
- DoD software with coding done in foreign environments or by foreign nationals is reviewed for malicious code by software quality assurance personnel;
- When employing commercial, off-the-shelf (COTS) software, preference is given during product selection and evaluation to those vendors who can demonstrate that they took efforts to minimize the security risks associated with foreign nationals who developed, modified, or remediated the COTS software being offered; and
- Software quality assurance personnel review software sent to locations not directly controlled by the DoD or its contractors for malicious code when it is returned to the DoD contractor's facilities.

4.4.5. Commercial-off-the-Shelf Items (COTS)

Use of commercial items offers significant opportunities for reduced development time faster insertion of new technology, and lower life cycle costs, owing to a more robust industrial base. No matter how much of a system is provided by commercial items, the program manager still should engineer, develop, integrate, test, evaluate, deliver, sustain, and manage the overall system. Particular attention should be paid to the intended usage environment and understanding the extent to which this differs from (or is similar to) the commercial usage environment; subtle differences in usage can have significant impact on system safety, reliability, and durability.

Technology risk considerations should receive intensive consideration when COTS are evaluated as the system concept is developed. Maximum use of mature technology provides the greatest opportunity to hold fast to program cost, schedule, and performance requirements and is consistent with an evolutionary acquisition strategy.

When acquiring COTS software products or other commercial items, the program manager still implements a robust systems engineering process. In this context, integration encompasses the amalgamation of multiple COTS components into one deployable system or the assimilation of a single COTS product (such as an enterprise resource planning system). In either case, the program manager should ensure that the system co-evolves with essential changes to doctrine (for combat systems) or reengineered business processes (for combat support and information technology systems) and apply commercial item best practices in the following areas:

- Adapting to commercial business practices;
- COTS evaluation;

- Relationship with vendors;
- Life-cycle planning; and
- Test and evaluation (T&E) of COTS items.

Adapting to Commercial Business Practices. When purchasing a commercial item, the program manager should adopt commercial business practice(s). The extent to which the DoD business practices match the business practices supported by commercial items determines the likelihood that the items will meet DoD needs, yet still realize the intended cost savings. It is likely, however, that a gap will exist—and the gap may be large. Negotiation, flexibility, and communication on the part of the stakeholders, the commercial vendors, and the program manager are required.

COTS Evaluation. The program manager should plan for and implement robust evaluations to assist in fully identifying commercial capabilities, to choose between alternate architectures and designs, to determine whether new releases continue to meet requirements, and to ensure that the commercial items function as expected when linked to other system components. In addition, evaluation provides the critical source of information about the trade studies that should be made between the capabilities of the system to be fielded and the system architecture and design that makes best use of commercial capabilities. Evaluating commercial items requires a focus on mission accomplishment and matching the commercial item to system requirements.

For COTS software, program managers are encouraged to use code-scanning tools, within the scope and limitations of the licensing agreements, to ensure both COTS and Government off-the-shelf software do not pose any information assurance or security risks. Section 7.10 of this Guidebook discusses the considerations for COTS software solutions.

Life-Cycle Planning. The program manager should establish a rigorous change management process for life-cycle support. Systems that integrate multiple commercial items require extensive engineering to facilitate the insertion of planned new commercial technology. This is not a "one time" activity because unanticipated changes may drive reconsideration of engineering decisions throughout the life of the program. Failure to address changes in commercial items and the marketplace will potentially result in a system that cannot be maintained as vendors drop support for obsolete commercial items.

Relationship with Vendors. The program manager needs to remain aware of and influence product enhancements with key commercial item vendors to the extent practical and in compliance with Federal Advisory Committee Act. As vendors are different from contractors and subcontractors, different practices and relationships are needed. Vendors react to the marketplace, not the unique needs of DoD programs. To successfully work with vendors, the program manager may need to adopt practices and expectations that are similar to other buyers in the marketplace. Traditional DoD acquisition and business models are not sufficient for programs acquiring commercial items, as they do not take into account the marketplace factors that motivate vendors.

T&E of COTS Items. The program manager should develop an appropriate test and evaluation strategy for commercial items to include evaluating potential commercial items in a system test bed, when practical; focusing test beds on high-risk items; and testing commercial-item upgrades for unanticipated side effects in areas such as security, safety, reliability, and performance.

4.4.6. Manufacturing Capability

4.4.6.1. Producibility

Producibility is the degree to which the design of the system facilitates the timely, affordable, and optimum-quality manufacture, assembly, and delivery of the system to the customer and should be a development priority. Design engineering efforts concurrently develop producible and testable designs, capable manufacturing processes, and the necessary process controls to satisfy requirements and minimize manufacturing costs. The program manager should use existing manufacturing processes whenever possible. When the design requires new manufacturing capabilities, the program manager needs to consider process flexibility (e.g., rate and configuration insensitivity).

Full rate production of a system necessitates a stable design, proven manufacturing processes, and available or programmed production facilities and equipment.

4.4.6.2. Manufacturing Readiness Levels

Engineering and Manufacturing Readiness Levels are a means of communicating the degree to which a technology is producible, reliable, and affordable. Their use is consistent with efforts to include the consideration of engineering, manufacturing and sustainment issues early in a program. More information can be found in the *Manager's Guide to Technology Transition in an Evolutionary Acquisition Environment*. Application of EMRLs should be tightly integrated with the technical reviews detailed in Section 4.3.

4.4.7. Quality

The quality of products, or services is determined by the extent they meet (or exceed) requirements and satisfy the customer(s), at an affordable cost. Quality is a composite of material attributes, including performance and product/service features and characteristics that satisfy a customer's requirement. A key to success is to systems engineer/design quality into the product by defining the product or service quality requirements from the beginning and then providing the contractor with the maximum degree of flexibility to meet these requirements.

The contractor is responsible for the quality of its products. The program manager should allow contractors to define and use their preferred quality management process that meets required program support capabilities. International quality standards ISO 9001–2000, *Quality Management Systems – Requirements*, or AS 9100:2001, *Quality Management Systems – Aerospace Requirements*, define process-based quality management systems and are acceptable for use on contracts for complex or critical items per FAR 46.202-4, *Higher-Level Contract Quality Requirements* < http://farsite.hill.af.mil/vffara.htm >.

A contractor's quality management system should be capable of the following key activities:

- Monitor, measure, analyze, control, and improve processes;
- Reduce product variation;

- Measure/verify product conformity;
- Establish mechanisms for field product performance feedback; and
- Implement an effective root-cause analysis and corrective action system.

Many companies pursue quality certification of their quality management systems as a goal in itself, rather than setting continuous quality improvement as a goal or using their quality management systems to help develop capable processes. There have been instances where a supplier has been ISO 9001 certified and the supplier's product was deficient or life threatening. The program manager should not require ISO certification of a supplier's quality program. Certification is just one means that a program manager uses to distinguish between multiple bidders. Past performance is another example. Contractors that achieve Six Sigma quality in their production processes would require the bare minimum quality oversight for acceptance of their products by the Government.

4.4.8. Reliability, Availability and Maintainability (RAM)

The program manager should establish RAM objectives early in the acquisition cycle and address them as a design parameter throughout the acquisition process. The program manager develops RAM system requirements based on the ICD or CDD and total ownership cost (TOC) considerations, and states them in quantifiable, operational terms, measurable during DT&E and OT&E. RAM system requirements address all elements of the system, including support and training equipment, technical manuals, spare parts, and tools. These requirements are derived from, and support, the user's system readiness objectives. Reliability requirements address mission reliability and logistics reliability. The former addresses the probability of carrying out a mission without a mission-critical failure. The latter is the ability of a system to perform as designed in an operational environment over time without any failures. Availability requirements address the readiness of the system. Availability is a function of the ability of the system to perform without failure (reliability) and to be quickly restored to service (a function of both maintainability and the level and accessibility of support resources). Maintainability requirements address the ease and efficiency with which servicing and preventive and corrective maintenance can be conducted; i.e., the ability of a system to be repaired and restored to service when maintenance is conducted by personnel of specified skill levels and prescribed procedures and resources.

Application of RAM and producibility activities during design, development, and sustainment is guided by a concise understanding of the concept of operations, mission profiles (functional and environmental), and desired capabilities. Such understanding is invaluable to understanding the rationale behind RAM and producibility activities and performance priorities. In turn, this rationale paves the way for decisions about necessary trade studies between system performance, availability, and system cost, with impact on the cost effectiveness of system operation, maintenance, and logistics support. The focus on RAM should be complemented by emphasis on system manufacturing and assembly, both critical factors related to the production and manufacturing, and to the sustainment cost of complex systems.

The program manager plans and executes RAM design, manufacturing development, and test activities so that the system elements, including software, that are used to demonstrate system performance before the production decision reflect a mature design. IOT&E uses production representative systems, actual operational procedures, and personnel with representative skill levels. To reduce testing costs, the program manager should utilize M&S in the demonstration of RAM requirements, wherever appropriate. (See DoD 3235.1-H.)

An additional challenge associated with RAM is the stochastic nature of the performance parameter. Typically, a large proportion of system requirements is deterministic and can be easily and repeatedly measured; e.g., the weight of an item is easily measured and can be repeated on a consistent basis. By contrast, a test of the reliability of an item is an evaluation of a sample, from which the population performance is inferred. The item may be performing to its average reliability requirement as specified, but the sample may return a higher or lower value. Repeated or more extensive samples would provide greater information about the underlying performance. The true reliability of the item is never really known until the item has completed its service. Until that point, the performance may be sampled, and confidence bounds determined for the population performance. Development of RAM requirements and the associated demonstration methods need to consider the stochastic nature of these parameters.

4.4.9. Supportability

The program manager should conduct supportability activities throughout the system life cycle. When using an evolutionary acquisition strategy, supportability activities address performance and support requirements for both the total life cycle of the system and for each capability increment, and consider and mitigate the impact of system variants or variations. The supportability of the design(s) and the acquisition of systems should be cost-effective and provide the necessary infrastructure support to achieve peacetime and wartime readiness requirements. Supportability considerations are integral to all trade-off decisions, as required in DoDI 5000.1, E1.17:

> *PMs shall consider supportability, life cycle costs, performance, and schedule comparable in making program decisions. Planning for Operation and Support and the estimation of total ownership costs shall begin as early as possible. Supportability, a key component of performance, shall be considered throughout the system life cycle.*

Supportability is the inherent quality of a system - including design for reliability and maintainability, technical support data, and maintenance procedures - to facilitate detection, isolation, and timely repair/replacement of system anomalies. This includes factors such as diagnostics, prognostics, real-time maintenance data collection, ‘design for support’ and ‘support the design’ aspects, corrosion protection and mitigation, reduced logistics footprint, and other factors that contribute to optimum environment for developing and sustaining a stable, operational system. To minimize the logistics footprint, the supportability posture of defense systems should be designed-in. The “footprint problem” has an engineering solution.

4.4.9.1. Supportability Analyses

The program manager conducts supportability analyses as an integral part of the systems engineering process throughout the system life cycle. The results of these analyses form the basis for the related design requirements included in the system performance specification and in the documentation of logistics support planning. The results also support subsequent decisions to achieve cost-effective support throughout the system life cycle. For systems, this includes all increments of new procurements and major modifications and upgrades, as well as reprocurement of systems, subsystems, components, spares, and services that are procured beyond the initial production contract award. The program manager should permit broad flexibility in contractor proposals to achieve program supportability objectives.

4.4.9.2. Support Concepts

The program manager establishes logistics support concepts (e.g., organic, two-level, three-level, contractor, partnering) early in the program, and refines the concepts throughout program development. Total ownership cost plays a key role in the overall selection process. Support concepts for all systems provide cost effective, total-life-cycle, logistics support.

Support concepts include the following:

- Embedded Diagnostics and Prognostics;
- Embedded Training and Testing;
- Serialized Item Management;
- Automatic Identification Technology;
- Iterative Technology Refreshment;
- Data Syntax and Semantics; and
- Unique Identification.

4.4.9.3. Support Data

Contract requirements for deliverable support and support-related data should be consistent with the planned support concept and represent the minimum essential requirements to cost-effectively maintain the fielded system and foster source of support competition throughout the life of the fielded system. The program manager coordinates Government requirements for this data across program functional specialties to minimize redundant contract deliverables and inconsistencies.

4.4.9.4. Support Resources

The support resources needed, for both the total system over its expected life and for each increment of introduced capability, are inherent to “full funding” calculations. Therefore, support resource requirements are a key element of program reviews and decision meetings. During program planning and execution, logistics support products and services are competitively sourced. The program manager should consider embedded training and maintenance techniques to enhance user capability and reduce life-cycle costs.

The program manager generally uses automatic test system (ATS) families or COTS components that meet defined ATS capabilities to meet all acquisition needs for automatic test equipment hardware and software. Critical hardware and software elements define ATS capabilities. The program manager considers diagnostic, prognostic, system health management, and automatic identification technologies and bases ATS selection on a cost and benefit analysis over the complete system life cycle. Consequently, the program manager is seeking to minimize the introduction of unique types of ATS into the DoD field, depot, and manufacturing operations.

4.4.10. Human Systems Integration (HSI)

Per DoD Directive 5000.1, the program manager shall pursue HSI initiatives to optimize total system performance and minimize total ownership cost. To do this, the program manager shall work with the manpower, personnel, training, safety, and occupational health, habitability, survivability, and human factors engineering (HFE) communities to translate and integrate the HSI thresholds and objectives contained in the capabilities documents into quantifiable and measurable system requirements (see DoD Instruction 5000.2). The program manager then includes these requirements in specifications, the Test and Evaluation Master Plan (TEMP), and other program documentation, as appropriate, and uses them to address HSI in the statement of work and contract. The program manager identifies any HSI-related schedule or cost issues that could adversely impact program execution; the system's support strategy should identify responsibilities, describe the technical and management approach for meeting HSI requirements, and summarize major elements of the associated training system (see 6.4.5.2.1.). See also MIL STD 1472F, Human Engineering. HSI topics include:

- Human Factors Engineering (DoD Instruction 5000.2 and Guidebook section 6.3);
- Habitability and Personnel Survivability (DoD Instruction 5000.2 and Guidebook sections 4.4.2, 6.2.6, 6.2.7.);
- Manpower Initiatives (DoD Instruction 5000.2 and Guidebook section 6.2.1);
- Personnel Initiatives (DoD Instruction 5000.2 and Guidebook section 6.2.2); and
- Training (DoD Instruction 5000.2, DoD Directive 1430.13, *Training Simulators and Devices,* and Guidebook section 6.2.3).

4.4.11. Environment, Safety and Occupational Health (ESOH)

As part of the program's overall cost, schedule, and performance risk reduction, the program manager shall prevent ESOH hazards, where possible, and manage ESOH hazards where they cannot be avoided (see 6.2.4.1, 6.2.5.2., and 6.2.5.3.). More specifically, DoD Instruction 5000.2 establishes requirements for program managers to manage ESOH risks for their system's life cycle. The program manager is required to have a PESHE document at MS B that describes

- The strategy for integrating ESOH considerations into the systems engineering risk management process using the methodologies described in the government-industry standard, *Standard Practice for System Safety*, MIL-STD-882D;
- The schedule for completing the National Environmental Policy Act (NEPA) (42 U.S.C. 4321-4370d) and Executive Order 12114 documentation;
- The status of ESOH risks management. The Acquisition Strategy, includes a summary of the PESHE;
- From MS B on, the PESHE document serves as a repository for top-level management information on ESOH risk; and
- Identification, assessment, mitigation, residual risk acceptance, and on-going evaluations of mitigation effectiveness and on NEPA compliance.

The ESOH systems engineering activities are described in further detail in the following sections:

- Programmatic Environment, Safety, and Occupational Health Evaluation (PESHE)
- ESOH Risk Management

Additional detailed guidance, processes and tools are available at the ESOH Special Interest Area on the Acquisition Community Connection web site.

4.4.11.1. Programmatic Environment, Safety, and Occupational Health Evaluation (PESHE)

There is no specific format for the PESHE. The program manager documents the PESHE in whatever manner is most useful to the program and best communicates to decision makers what ESOH issues affect the program. The PESHE transitions from an initial planning document at MS B into an ESOH risk management tool as the program matures.

The PESHE includes the following:

- Strategy for integrating ESOH considerations into the systems engineering process
- Identification of who is responsible for implementing the ESOH strategy
- Approach to identifying ESOH risks, reducing or eliminating the risks, and implementing controls for managing those ESOH risks where the program cannot avoid them;
- Identification, assessment, mitigation, and acceptance of ESOH risks. DoD Instruction 5000.2, E7.7 establishes the acceptance authorities for residual risks as: the DoD Component Acquisition Executive for high risks, the Program Executive Office-level for serious risks, and the program manager for medium and low risks as defined in MIL-STD-882D;
- Method for tracking progress in the management and mitigation of ESOH risks and for measuring the effectiveness of ESOH risk controls;
- Compliance schedule for completing National Environmental Policy Act (NEPA)/ Executive Order 12114 documentation;
- Identification of hazardous materials (HAZMAT), including energetics, used in the system;
- Approach for, and progress in, integrating HAZMAT, energetics, and other ESOH considerations (e.g., environmental impacts, personnel safety, regulatory compliance) into system demilitarization and disposal planning (see 4.4.14); and
- Approach for, and progress in, integrating ESOH into test and evaluation (T&E) planning and reporting.

DoD Instruction 5000.2 does not require that the PESHE supersede or replace other ESOH plans, analyses, and reports (e.g., System Safety Management Plan/Assessments, HAZMAT Management Plan, Pollution Prevention Plan, Health Hazard Assessments, etc.); the program manager incorporates these documents by reference, as appropriate. However, to the maximum extent possible, the program manager should minimize duplication of effort and documentation and give preference to recording ESOH information in the PESHE, as opposed to maintaining a series of overlapping, redundant documents. Human Systems Integration also addresses many of the safety and health ESOH areas. The PESHE describes the linkage between ESOH and HSI and how the program avoids duplication of effort.

The required compliance schedule for completing NEPA/E.O. 12114 documentation, as detailed in the PESHE and summarized in the Acquisition Strategy, includes the following:

- Events or proposed actions (to include T&E and fielding/basing activities) throughout the life cycle of the program that may require preparation of formal NEPA documentation
- Proponent for each proposed action having the lead to prepare the formal NEPA documentation
- The anticipated initiation date for each proposed action
- The anticipated type of NEPA/E.O. 12114 document (e.g., Categorical Exclusion, Environmental Assessment and Finding of No Significant Impact, or Environmental Impact Statement and Record of Decision) which the proponent should complete prior to the proposed action start date
- The anticipated start and completion dates for the final NEPA/E.O. 12114 document
- The specific approval authority for the documents. DoD Instruction 5000.2, E7.7 establishes the DoD Component Acquisition Executive or designee (for joint programs, the DoD Component Acquisition Executive of the Lead Executive DoD Component) as the approval authority for system-related NEPA/E.O. 12114 documentation.

4.4.11.2. ESOH Risk Management

Balancing the elimination or reduction of Environment, Safety and Occupational Health (ESOH) risk with an informed and structured residual risk acceptance process is essential for positively contributing to a program's efforts in meeting cost, schedule, and performance requirements. ESOH risks are part of each program’s overall cost, schedule, and performance risks, and the program manager should review them from within that overall context. Risk acceptance and implementation of effective mitigating measures/controls is necessary to avoid loss of life or serious injury to personnel; serious damage to facilities or equipment resulting in large dollar loss; failures with adverse impact on mission capability, mission operability, or public opinion; and harm to the environment and the surrounding community.

The ESOH risk management process uses ESOH risk analysis matrices, based on the guidance in MIL-STD-882D. The risk matrices should use clearly defined probability and severity criteria (either qualitative or quantitative) to categorize ESOH risksProgram managers elect to either establish a single consolidated ESOH risk matrix or use individual environmental, safety, and occupational health matrices.

The three basic types of ESOH risks are

- Potential ESOH impacts and adverse effects from routine system development, testing, training, operation, sustainment, maintenance, and demilitarization/disposal;
- Potential ESOH and mission readiness impacts from system failures or mishaps, including critical software failures; and

- Potential impacts to program life-cycle cost, schedule, and performance from ESOH compliance requirements.

The scope of potential risks includes all ESOH regulatory compliance requirements associated with the system throughout its life cycle, such as, but not limited to, the following:

- HAZMAT use and hazardous waste generation;
- Demilitarization and disposal requirements;
- Safety (including explosives safety, ionizing and non-ionizing radiation);
- Human health (associated with exposure to chemical, physical, biological, or ergonomic hazards, etc.);
- Environmental and occupational noise; and
- Impacts to the natural environment (e.g., air, water, soil, flora, fauna).

ESOH risk information should include the following:

- Description of the risk/hazard;
- Preliminary risk assessment;
- Necessary mitigation measures to eliminate or reduce the risk;
- Residual risk assessment;
- Residual risk acceptance document; and
- Mitigation measure effectiveness.

Programs begin the process of identifying ESOH risks using lessons learned from the following sources of information:

- Legacy systems that the new system will replace, to include mishap and lost time rates associated with any legacy system;
- Similar systems;
- Pre-system acquisition activities (e.g., the Technology Development Strategy);
- Demilitarization and disposal of similar systems; and
- ESOH regulatory issues at potential locations for system testing, training, and fielding/basing.

In addition to standard ESOH risk management data, HAZMAT (to include energetics) risk information includes:

- The locations and quantities of HAZMAT on the system, where applicable;
- Energetic qualification information for each energetic material used in the system;
- Reasonably anticipated hazardous byproducts/discharges and expected quantities of hazardous waste generated during normal use/maintenance, in addition to those anticipated in emergency situations (e.g., exhaust, fibers from composite materials released during accidents, etc.);
- Special HAZMAT training and handling, and Chemical, Biological, Radiological, and Nuclear contamination.

The preferred mitigation strategy is source reduction or elimination of the hazards, also referred to as pollution prevention when dealing with potential environmental impacts. The program manager should strive to eliminate or reduce ESOH risks as part of the system's total life-cycle risk reduction strategy. For systems containing energetics, source reduction consists of minimizing the use of the energetic materials and developing system designs that reduce the possibility and consequences of an explosive mishap. This includes complying with the insensitive munitions criteria (per DoD Directive 5000.1) and pursuing hazard classifications and unexploded ordnance liabilities that minimize total ownership cost (see section 4.4.16).

If effectively executed, ESOH risk management sets the stage for addressing National Environmental Policy Act (NEPA)/Executive Order 12114 requirements by identifying system-specific ESOH risk information. The program manager combines these data with the geographic/site specific environmental conditions and requirements, to prepare formal NEPA analysis documents. In addition, the program manager is responsible to provide system specific ESOH risk data in support of NEPA analysis by other Action Proponents. This approach streamlines the overall NEPA/E.O. 12114 analysis process, reducing cost and schedule impacts. The program manager should integrate into the ESOH risk management data any additional ESOH risks or additional mitigation measures identified during the formal NEPA/E.O. 12114 analysis process.

The program manager should monitor and assess the effectiveness of mitigation measures (i.e., tracking ESOH progress in terms of regulatory compliance) to determine whether additional control actions are required. The program manager then documents the effectiveness of mitigation measures in the PESHE. Relevant information can include any related mishap data, adverse health effects, and significant environmental impacts from system development, testing, training, operation, sustainment, maintenance, and demilitarization/disposal. Programs can also convey information about the effectiveness of their risk management efforts with metrics, achievements, success stories, etc.

4.4.12. Survivability and Susceptibility

The program manager should fully assess system and crew survivability against all anticipated threats at all levels of conflict early in the program, but in no case later than entering System Demonstration and Demonstration. This assessment also considers fratricide and detection. If the system or program has been designated by the Director, Operational Test and Evaluation (DOT&E), for Live Fire Test and Evaluation (LFT&E) oversight, the program manager should integrate the test and evaluation (T&E) used to address crew survivability issues into the LFT&E program supporting the Secretary of Defense LFT&E Report to Congress.

The program manager should address Nuclear, Biological and Chemical and High Altitude Electromagnetic Pulse cost-effective survivability techniques and plan for the validation and confirmation of NBC and HEMP survivability.

The program manager should establish and maintain a survivability program throughout the system life cycle to attain overall program objectives. The program should stress early investment in survivability enhancement efforts that improve system operational readiness and mission effectiveness by:

- Providing threat avoidance capabilities (low susceptibility);

- Incorporating hardening and threat tolerance features in system design (low vulnerability)
- Providing design features to reduce personnel casualties resulting from damage to or loss of the aircraft (casualty reduction)
- Maximizing wartime availability and sortie rates via operationally compatible threat damage tolerance and rapid reconstitution (reparability) features
- Minimizing survivability program impact on overall program cost and schedule
- Ensuring protection countermeasures and systems security applications are defined for critical component's vulnerability to validated threats for systems survivability, including conventional or nuclear advanced technology weapons; nuclear, biological, or chemical contamination; and electronic warfare threats

Unless waived by the Milestone Decision Authority, mission-critical systems, including crew, regardless of acquisition category, should be survivable to the threat levels anticipated in their projected operating environment as portrayed in the System Threat Assessment. Design and testing ensure that the system and crew can withstand man-made hostile environments without the crew suffering acute chronic illness, disability, or death.

The program manager should ensure that system susceptibility is addressed as a design consideration. Electromagnetic compatibility (EMC) and electromagnetic interference (EMI) should be addressed against the planned operational environment and the effects it may have on the system. Additionally, EMC/EMI should be a consideration within the system to understand unintended electromagnetic coupling across and among system components under various operational and maintenance scenarios. MIL-STD-461 or similar procedures can provide a basis for the technical design and certification approach for EMC/EMI. Section 7.6 contains additional detail about spectrum management considerations.

4.4.13. Corrosion Prevention and Control

The program manager should consider and implement corrosion prevention and mitigation planning to minimize the impact of corrosion and material deterioration throughout the system life cycle (see the *Corrosion Prevention and Control Planning Guidebook*). Corrosion prevention and mitigation methods include, but are not limited to, the use of effective design practices, material selection, protective finishes, production processes, packaging, storage environments, protection during shipment, and maintenance procedures. The program manager establishes and maintains a corrosion prevention and mitigation reporting system for data collection and feedback and uses it to adequately address corrosion prevention and mitigation logistic considerations and readiness issues. Corrosion prevention and mitigation considerations are integral to all trade-off decisions for Performance Based Logistics (see section 5.3.) as required in DoD Directive 5000.1:

> *Performance-Based Logistics. PMs shall develop and implement performance-based logistics strategies that optimize total system availability while minimizing cost and logistics footprint. Trade-off decisions involving cost, useful service, and effectiveness shall consider corrosion prevention and mitigation. Sustainment strategies shall include the best use of public and private sector capabilities through government/industry partnering initiatives, in accordance with statutory requirements.*

4.4.14. Disposal and Demilitarization

During systems engineering as part of the program manager's total life cycle systems management (TLCSM) responsibilities, the program manager should consider materiel demilitarization and disposal. The program manager should coordinate with DoD Component logistics and explosive safety activities and the Defense Logistics Agency, as appropriate, to identify and apply applicable demilitarization requirements necessary to eliminate the functional or military capabilities of assets (DoD 4140.1-R and DoD 4160.21-M-1) and to determine reutilization and hazardous-property disposal requirements for system equipment and by-products (DoD 4160.21-M).

For a munitions program, the program manager shall document the parts of the system that will require demilitarization and disposal and addresses the inherent dangers associated with ammunition and explosives (DoD Instruction 5000.2). This documentation should be in place before the start of developmental test and evaluation and before the program manager releases munitions or explosives to a non-military setting. The documentation provides the following:

- Render safe procedures—step-by-step procedures for disassembling the munitions item(s) to the point necessary to gain access to or to remove the energetic and hazardous materials; and
- Identification of all energetics and hazardous material, and the associated waste streams produced by the preferred demilitarization/disposition process.

Open burn and open detonation are not to be considered as the primary methods of demilitarization or disposal.

4.4.15. Information Assurance (IA)

The program manager) should incorporate information assurance requirements into program design activities to ensure availability, integrity, authentication, confidentiality, and non-repudiation of critical system information (see DoD Directive 5000.1). DoD policy for information assurance of information technology, including National Security Systems (NSS), appears in DoD Directive 8500.1, *Information Assurance (IA)* and implementing instructions in DoD Instruction 8500.2, *Information Assurance (IA)*. Because the requirements for IA vary greatly across acquisition programs, it is essential that a program manager examine his/her acquisition program carefully to identify applicable IA requirements. Sections 7.5 and 8.3.3 of this Guidebook provide additional guidance on the extent and elements of IA that should be considered.

4.4.16. Insensitive Munitions

The ultimate objective when making design decisions on munitions is to develop and field munitions that have no adverse reaction to unplanned stimuli. All munitions and weapons, regardless of Acquisition Category (ACAT) level, should conform to insensitive munitions (unplanned stimuli) criteria and use materials consistent with safety and interoperability requirements. The Joint Capabilities Integration and Development System validation process determines insensitive munitions requirements and keeps them current throughout the acquisition cycle. Munitions insensitivity is certified per CJCS Instruction 3170.01. Waivers for munitions/weapons, regardless of ACAT level, require Joint Requirements Oversight Council (JROC) approval.

All submunitions and weapon submunitions, regardless of ACAT, should conform to the policy of reducing overall unexploded ordnance through a process of improving the submunitions system reliability – the desire is to field future submunitions with a 99% or higher functioning rate (SecDef Memorandum, 10 Jan 01, subject: DoD Policy on Submunition Reliability). The JROC approves any waivers for this policy for "future" ACAT I and II submunitions weapons programs. A future submunitions weapon is one that will reach Milestone C in fiscal year 2005 and beyond.

4.4.17. Anti-Tamper Provisions

Anti-tamper activities encompass the system engineering activities intended to prevent or delay exploitation of critical technologies in U.S. systems. These activities involve the entire life cycle of systems acquisition, including research, design, development, testing, implementation, and validation of anti-tamper measures. Properly employed, anti-tamper measures will add longevity to a critical technology by deterring efforts to reverse-engineer, exploit, or develop countermeasures against a system or system component.

The program manager should develop and implement anti-tamper measures in accordance with the determination of the Milestone Decision Authority (MDA), as documented in the anti-tamper annex to the program protection plan (see DoD 5200.1-M, *Acquisition Systems Protection Program*). Anti-tamper capability, if determined to be required for a system, is reflected in the systems specifications, integrated logistics support plan, and other program documents and design activities. Because of its function, anti-tamper should not be regarded as an option or a system capability that may later be traded off without a thorough operational and acquisition risk analysis. To accomplish this, the program manager identifies critical technologies and system vulnerabilities and, with assistance from counter-intelligence organizations, performs threat analyses on the critical technologies. Additionally, the program manager researches anti-tamper measures and determines which best fit the performance, cost, schedule, and risk of the program.

The program manager should also plan for post-production anti-tamper validation of end items. The Department’s anti-tamper executive agent may develop and execute a validation plan and report results to the MDA and Component Acquisition Executive.

4.4.18. System Security

The program manager should consider security, survivability, and operational continuity (i.e., protection) as technical performance parameters as they support achievement of other technical performance aspects such as accuracy, endurance, sustainability, interoperability, range, etc., as well as mission effectiveness in general. The program manager includes these considerations in the risk benefit analysis of system design and cost. Users are familiar with critical infrastructure protection and space control requirements, and account for necessary hardening, redundancy, backup, and other physical protection measures in developing system and system-of-systems capability documents and architectures.

4.4.18.1. Research and Technology Protection (RTP)

A component of overall system security, research and technology protection identifies and safeguards selected DoD research and technology anywhere in the Research, Development, Test and Evaluation or acquisition processes to include associated support systems (e.g., test and simulation equipment). This involves integrating all security disciplines, counterintelligence, intelligence, and other defense methods to protect critical science and technology from foreign collection or unauthorized (see also Chapter 8).

4.4.18.2. System Security Engineering (SSE)

System security engineering is an important element of Research and Technology Protection (RTP) and the vehicle for integrating RTP into a system during the design process. Not only does security engineering address potential unauthorized collection or disclosure, it also considers the possible capture of the system by an adversary during combat or hostile action and what security countermeasures are important during design to prevent reverse engineering. A discretionary Systems Security Management Plan documents recommended formatting, contents, and procedures for the SSE manager and contractors implementing SSE. Guidance for SSE assessments and preparation of the SSE management plan are contained in Military Handbook 1785, *System Security Engineering.*

4.4.19. Accessibility

The program manager must ensure that electronic and information technology acquisitions comply with Section 508 of the Rehabilitation Act of 1973. Unless an exception at Federal Acquisition Regulation 39.204 applies, acquisitions of electronic and information technology supplies and services must meet the applicable accessibility standards at Title 36 Code of Federal Regulations Section 1194. To avoid unnecessary costs and delays, the program manager should consider what accessibility requirements, if any, are applicable to the program early and throughout the system life cycle.

4.4.20. Unique Identification of Items

DoD Unique Identification (UID) permanently identifies an individual item. The serialized item is then distinct from all other individual items that the DoD buys or owns. With UID, the DoD can associate valuable business intelligence to an item throughout its life cycle. The UID system accurately captures and maintains data for valuation and tracking of items.

The DoD UID program places a minimum set of globally unique and unambiguous data markings on each identified item. The robust system ensures data integrity throughout the life of the item, and support multi-faceted business applications and users.

The following sources provide useful information about UID:

- An Acting Under Secretary of Defense (Acquisition, Technology, and Logistics) Memorandum dated July 29, 2003. The memo contains the basic UID requirements and makes UID a mandatory requirement for all solicitations issued on or after 1 January 2004 by the Department.
- A DoD UID guide containing Frequently Asked Questions and a set of UID business rules, available at http://www.acq.osd.mil/uid.
- A DFARS interim rule regarding UID (DFARS Case 2003-D081)

- *Guide to Uniquely Identifying Items* that specifies Identification Marking of U.S. Military Property.

4.5. Systems Engineering Execution: Key Systems Engineering Tools and Techniques

This section describes many of the systems engineering techniques and tools for management, oversight, and analysis and provides some general knowledge management resources.

4.5.1. Systems Engineering Plan

The Systems Engineering Plan (SEP) is a detailed formulation of actions that should guide all technical aspects of an acquisition program. Program managers should establish the SEP early in program formulation and update it at each subsequent milestone. It is intended to be a living document, tailored to the program, and a roadmap that supports program management by defining comprehensive systems engineering activities, addressing both government and contractor technical activities and responsibilities. This chapter of the Guidebook, in its entirety, should be taken as guidance for preparation of a SEP.

The SEP describes the program's overall technical approach, including systems engineering processes; resources; and key technical tasks, activities, and events along with their metrics and success criteria. Integration or linkage with other program management control efforts, such as integrated master plans, integrated master schedules, technical performance measures, and earned value management, is fundamental to successful application.

There is no prescribed format for the SEP. However, it should address how systems engineering will support the translation of system capability needs into an effective, suitable product that is sustainable at an affordable cost. Specifically, a well-prepared SEP will address the integration of the technical aspects of the program with the overall program planning, systems engineering activities, and execution tracking to include:

- The systems engineering processes to be applied in the program (e.g., from a standard, a capability maturity model, or the contractor's process). Describe how the processes will be implemented and how they will be tailored to meet individual acquisition phase objectives. Describe how the systems engineering processes will support the technical and programmatic products required of each phase. Sections 4.2 (process) and 4.3 (process application to SE phase) provide a "roadmap" of how SE processes can be applied to an acquisition program.
- The system's technical baseline approach. Describe how the technical baseline will be developed, managed, and used to control system requirements, design integration, verification, and validation. Include a discussion of metrics (e.g., technical performance measures) for the technical effort and how these metrics will be used to measure progress.
- Event-driven timing, conduct, success criteria, and expected products of technical reviews, and how technical reviews will be used to assess technical maturity, assess technical risk, and support program decisions. SEP updates shall include results of completed technical reviews. Section 4.3 of this guide, as well as other reference material on technical reviews, should form a basis for the program's approach.

- The integration of systems engineering into the program's integrated product teams (IPTs). Describe how systems engineering activities will be integrated within and coordinated across IPTs; how the IPTs will be organized; what SE tools they will employ; and their resources, staffing, management metrics, and integration mechanisms. Describe how systems engineering activities are integrated in the program's overall integrated schedules (4.5.2 and 4.5.3).
- For programs that are part of a system of systems or family of systems, the synchronization with related systems to achieve the desired mission capability as the system evolves. The relative contribution of each system to the overall mission capability in terms of performance and effectiveness should be identified to ensure that the combination of systems is appropriately integrated together.

In addition to describing required program activities, the SEP addresses the who, what, when, where, why, and how of the applied systems engineering approach.

Participants in the SE Process (Who) **–** Ideally, the SEP should detail roles and responsibilities of the systems engineering effort across the acquirer (government) and supplier (contractor) boundaries. Roles of the Chief Engineer, lead Systems Engineer, IPT SEs, Systems Engineering and Integration Teams, etc., need to be explicitly defined. Vertical and horizontal integration, team communications, and scope of decision-making authority are key elements of the plan, especially as these relate to management of technical baselines and reviews. SE staffing (planned vs. actual) should be included in this discussion together with (required vs. actual) discussion of domain experience of the staff.

SE Processes (What) **–** There are many ways to accomplish SE. Critical to the plan is which of these many ways will the program select and implement. There is a difference between complexity and uncertainty. While SE is complex, it should not be uncertain. The SEP should serve as a vehicle for minimizing process uncertainty. Optimally, a program team should use a single set of common SE processes. For large programs having multiple organizations, this may be an impractical goal. In these cases, the program manager should strive to "rationalize" or link the different process implementations across the program team so that process inputs and outputs integrate.

Facilities Enabling SE (Where) **–** The SEP should address development and use of SE facilities, including verification and validation facilities. Since these facilities can be complex hardware and software systems in their own right, the issue of integration facilities can be a significant challenge, particularly as relating to modeling and simulation development requirements.

SE Event Timing (When) **–** Systems engineering is an event-driven process. As such, the SEP should discuss the timing of events in relation to other SE and program events. While the initial SEP and Integrated Master Schedule will have the expected occurrence in the time of various milestones (such as overall system CDR), the plan should accommodate and be updated to reflect changes to the actual timing of SE activities, reviews, and decisions.

SE Decision Rationale (Why) **–** SE includes a continuous evolution of requirements (from high end to detail level) and trade offs (to best balance the design across often-conflicting design considerations). Rationale as to how these requirements and trades will be balanced should be included in the SEP. Decision criteria, such as entry and exit criteria for technical reviews, should be detailed.

Tools Enabling SE (How) -- Robust systems engineering makes use of a number of tools, toolsets, and enablers, such as modeling and simulation. The capability, variety, and dynamics of modern SE tools demand that they be fully integrated with the overall approach and discussion of SE application. Since adaptation of tools often occurs on programs, continual update of the SEP is required.

For programs where the USD(AT&L) is the Milestone Decision Authority, components shall submit the SEP at least 30 days before the scheduled Defense Acquisition Board milestone review. The MDA is the approval authority for the SEP (see USD(AT&L) SE Policy Memo of 20 Feb 04). The Director, Defense Systems, and members of the OSD staff will assess the SEP and other required milestone documents, identify and help resolve issues, and make a recommendation on the program's readiness to proceed to the Defense Acquisition Board.

4.5.2. Integrated Master Plan

The program manager should use event-driven schedules and the participation of all stakeholders to ensure that all tasks are accomplished in a rational and logical order and to allow continuous communication with customers. Necessary input conditions to complete each major task are identified, and no major task is declared complete until all required input conditions and component tasks have been satisfied. When documented in a formal plan and used to manage the program, this event-driven approach can help ensure that all tasks are integrated properly and that the management process is based on significant events in the acquisition life cycle and not on arbitrary calendar events.

One way of defining tasks and activities is the use of an integrated master plan, which provides an overarching framework against which all work is accomplished. It documents all the tasks required to deliver a high quality product and facilitate success throughout the product's life cycle. Cost, schedule (specific dates), and non-essential tasks are not included in this plan. During the initial stages of a program, the integrated plan is preliminary, and its purpose is to provide an understanding of the scope of work required and the likely structure of the program. It is constructed to depict a likely progression of work through the remaining phases, with the most emphasis on the current or upcoming phase (especially the period to be contracted for next). The integrated plan also serves to identify dependencies, which may be performed by different organizations.

As the program is defined, the integrated master plan is iterated several times, each time increasing the level of detail and confidence that all essential work has been identified. The specific format for this plan is not critical; however, it usually reflects an Event/Accomplishment/Criteria hierarchical structure—a format that greatly facilitates the tracking and execution of the program. Functional and life-cycle inputs are required to integrate the product and associated processes produced by the program. Without formal documentation, such as an integrated master plan, these inputs may be lost when personnel change. Such a plan also defines and establishes the correct expectations.

Deriving the program schedule presents an opportunity to identify critical risk areas. As the times to complete specific tasks are estimated, events that may cause delays will become apparent. These events are potential areas of risk that the program manager should consider for further analysis.

4.5.3. Integrated Master Schedule

Unlike event-based planning, time-based planning uses a calendar or detailed schedule to demonstrate how work efforts will support tasks and events. One way to produce such a schedule is to develop an integrated master schedule based on an integrated master plan. With an integrated master plan, the integrated master schedule further helps the program manager understand the links and interrelationships among the various teams. The integrated schedule begins as an integrated master plan with dates—the starting points are the events, accomplishments, and criteria that make up the plan. At a minimum, an integrated master schedule shows the expected start and stop dates for each criterion in the plan, but each criterion may be broken down into lower-level tasks that will be used to manage the program on a day-to-day basis. The schedule can be expanded downward to the level of detail appropriate for the scope and risk of the program. Programs with high risk show much lower levels of detail in the integrated master schedule in order to give the visibility to manage and control risk. The more detailed the integrated master schedule, however, the greater the cost to track and update the schedule. The dates in the integrated master schedule usually are not made contractually binding in order to allow the flexibility to take full advantage of event-driven scheduling.

Each of the work products requires different levels of effort, personnel, resources, and time to complete, with some being more difficult to complete than others. Critical Path Analysis is used to help identify which tasks, or sets of tasks, will be more difficult or costly to complete. As many of the tasks are inter-related and as work products typically require the completion of all lower level tasks before the higher-level work product can be completed, the early identification of critical tasks is essential for ensuring that schedule and cost goals are maintained for the program.

4.5.4. Value Engineering

The DoD value engineering program, per 41 U.S.C. 432, reduces cost, increases quality, and improves mission capabilities across the entire spectrum of DoD systems, processes, and organizations. It employs a simple, flexible, and structured set of tools, techniques, and procedures that challenge the status quo by promoting innovation and creativity. Furthermore, it incentivizes government participants and their industry counterparts to increase their joint value proposition in achieving best value solutions as part of a successful business relationship. Where appropriate, program managers should engage in a broad and rigorous application of the value engineering methodology. In addition, program managers should be receptive to Value Engineering Change Proposals (VECPs) made by contractors as a way of sharing cost savings and should also ensure that implementation decisions are made promptly.

4.5.5. Technical Performance Measurement

Systems engineering uses technical performance measurements to balance cost, schedule, and performance throughout the life cycle. Technical performance measurements compare actual versus planned technical development and design. They also report the degree to which system requirements are met in terms of performance, cost, schedule, and progress in implementing risk handling. Performance metrics are traceable to user-defined capabilities.

4.5.6. Trade Studies

Trade studies are conducted among operational capabilities, functional, and performance requirements, design alternatives and their related manufacturing, testing, and support processes; program schedule; and life-cycle cost. Such trade studies are made at the appropriate level of detail to support decision making and lead to a proper balance between system performance and cost. Requirements come from many sources and unfortunately can conflict with each other. Trade studies are used for the resolution of these conflicts.

4.5.7. Modeling and Simulation

As the Department of Defense continues its transformation, it increasingly relies on network centric operations and on individually-complex systems linked together in complex systems-of-systems. This transformation increases the dependency on seamless interoperability. Interoperability is needed between systems across military service and national boundaries, and requires effective performance by each individual system. The systems engineering process must exploit modeling and simulation to rapidly field improved capabilities with sufficient confidence that the fielded capabilities will perform effectively in the system-of-systems joint mission environment.

Modeling and simulation is an essential element of the systems engineering process. Modeling and simulation can represent the system-of-systems environment as a context for systems engineering to properly design, develop, and test individual systems. The cost and complexity of modern weapon systems, particularly within a family-of-systems or system-of-systems, preclude the development of full-scale prototypes to merely provide proof of concept. Similarly, the cost of testing events limits the number of tests that can be practically conducted. Modeling and simulation supports the systems engineering decision process by supporting systems design, trade studies, financial analysis, sustainment, and performance assessments.

The following paragraphs describe the contributions of modeling and simulation by phase.

4.5.7.1. M&S in Concept Refinement

A technical framework, including essential architecture products, is necessary for a program manager program manager to initiate the systems engineering process to allow interoperability with legacy, current, and future systems. M&S tools exist that can help define the technical framework to be part of the Capability Development Document. A prudent process includes development of a distributed collaborative environment accessible by all the stakeholders. M&S is a tool to support the collaborative process, to exchange data, consider alternatives (such as operational concepts, conceptual designs, cost, and technology strategies), and view potential resulting capabilities.

M&S will allow a program manager to conduct rapid virtual prototyping with all stakeholders playing a role in the system as part of a family-of-systems or systems-of-systems. A distributed collaborative environment will support authoritative information exchange and rapid refinement of the design or concept due to changing circumstances such as technological advancements and changing threats, tactics, or doctrine.

Characteristics of a collaborative environment will entail models and simulations at multiple locations that are run and operated by subject matter experts and connected by wide area networks on an as needed basis. As changes are made to define a system that meets the needed capability all stakeholders in the system's life-cycle will have an active role in the changes being made.

When a needed capability is identified, M&S can be used in the collaborative environment to examine and explore alternatives and variations to proposed concepts. Rigorous examination, by all of the stakeholders, to proposed and alternative concepts applied through the effective use of M&S can help identify enabling technologies, constraints, costs, and associated risks. This rigor early in the concept refinement process is vital because the resulting decisions made in this early phase have repercussions throughout the system's life-cycle that drive the ultimate life-cycle costs of the system.

Outputs of the concept refinement phase include the Systems Engineering Plan (SEP) which should include M&S support throughout the acquisition life-cycle and address M&S roles of both the government and industry. Of particular importance are configuration management, data rights and access, and responsibilities for life-cycle maintenance of data and models by industry and government. Appropriate standards to assure M&S interoperability and reuse of models and data should be addressed. Further, the test and evaluation (T&E) strategy should be defined with the role that M&S will play in augmenting and focusing the testing and evaluation process. Of vital importance is a strategy to continuously improve the veracity of the suite of M&S based on results from testing. The cyclical process of "model-test-fix-model" is applicable to assure M&S remains on the cutting edge of validity.

Key to successful simulation support to the systems engineering process is the recognition that M&S employed during the concept refinement stage can be leveraged throughout successive phases of the acquisition cycle. Ideally, the same architecture, scenarios, data, and M&S exercised in the collaborative environment during concept refinement will be reused in support of the analysis during the technology development.

4.5.7.2. M&S in Technology Development

M&S can be used during the Technology Development phase to help reduce technology risk and determine an appropriate set of technologies to integrate into a full system. With the establishment of the collaborative environment the same architecture, scenario, data, HWIL, SWIL, infrastructure, and some of the same M&S can be used to examine new technologies. M&S used in the development and demonstration of new technologies for Advanced Technology Demonstrations (ATDs) and Advanced Concept Technology Demonstrations (ACTDs) can be incorporated into the collaborative environment to determine how to interface the new technologies with legacy systems and determine the likelihood of their successful transition to support a needed capability.

A variety of M&S tools can be used to examine reliability, availability, maintainability, transportability, provisioning (spares, support equipment, manpower), cost implications, and human-machine interface design considerations for any new designs or applicable technologies that can be applied to specific capability needs. The program manager should make use of physics-of-failure and finite element analysis M&S for stress analysis, structural dynamics, mass properties, structural design materials, fatigue, loads, shock isolation, and acoustics. These M&S tools should be incorporated and made accessible through the established collaborative environment.

Cost models should also be employed to determine projected life-cycle costs of the system. As part of the cost estimate, M&S tools for manpower estimates can be employed. Alternatives to the traditional cost estimation techniques need to be considered because legacy cost models tend not to adequately address costs associated with information systems, FoS, and SoS.

Testing of new capabilities needs to include test and evaluation throughout the technology and system development process rather than solely relying on a single "pass-fail" test to move into production. The role of M&S in the testing process must be documented in the Test and Evaluation Master Plan (TEMP). With the assistance and proper application of M&S and the early coordination with operational testers, the operational tests can be integrated throughout the development process and incorporated with the developmental tests. As part of the developmental testing process, a program manager should identify data needed from the tests to further validate the M&S used in the collaborative environment.

Before hardware prototypes are built, virtual prototypes should be developed, evaluated, redesigned as appropriate, and then reevaluated. The "model-test-fix-model" process should be used under a spiral development paradigm to help identify an achievable capability with an ultimate goal of demonstrating capability in a virtual context before considering a hardware demonstration.

Outputs of the Technology Development phase include system performance specifications, the TEMP, an updated SEP, validated systems support, life-cycle cost estimates, and manpower requirements. M&S should play a significant role in all of these outputs during this phase of the acquisition process.

4.5.7.3. M&S in Systems Development and Demonstration

A key aspect of the systems development and demonstration phase includes the integration of the new technologies with legacy, current, and future systems. With the establishment of the architecture for the collaborative environment, many of the systems interface requirements should already be satisfied. This will be particularity true for any new systems developed utilizing the same architecture. In any case, M&S can be used in conjunction with HWIL, real world C4ISR systems, and other simulated systems to identify the required interface requirements in order to be an integral part of a family of systems or system of systems.

Verified and validated M&S, supported by validated test data, can be used to support the testing process to evaluate the performance and maturity of the technology under development. The program manager can make effective use of M&S to help focus T&E of hardware prototypes to maximize the highest pay off of the T&E investments. M&S can assist the T&E process by assessing a system in scenarios and areas of the mission space or performance envelope where testing cannot be performed, is not cost effective, or additional data is required. M&S must play a significant role in testing a system that is part of a family-of-systems or systems-of-systems. It is cost prohibitive and unrealistic to bring together all assets of a FoS or SoS to conduct live tests and evaluations of the systems' interactions. These systems interactions can however be examined in a simulated environment where all or selective assets of FoS or SoS can simulated.

Through the use of M&S, a system's capabilities and contributions to a FoS or SoS can demonstrated. Computerized representations of the system's human-machine interfaces can be provided to end-users to obtain final ergonomic modifications to the design. Making design changes in the computerized representations will be much less costly than making the same changes in hardware prototypes. Consideration should be given to using or modifying these same computerized representations to start training end-users on the new system. In such a simulated environment, final design trades and modifications can be made before going into production.

The M&S incorporated into the established collaborative environment supports transition to production phase. The digital design data associated with the system can be electronically transferred directly to the manufacturing floor minimizing ambiguity in the systems specifications.

4.5.7.4. M&S in Production and Development

The M&S used during the systems engineer process allows system designs to be electronically transmitted to the manufacturing shop floor to make the manufacturing process more streamlined and efficient. M&S can be used to not only produce detailed designs of a system; they can also be used to define the production and support processes for the system. M&S should be considered in designing manufacturing facilities, defining production flows to meet planned production rates, and eliminating production bottlenecks.

Before a new system goes into production, a program manager should examine the possibilities of modifying the computerized prototypes of the system to create virtual trainers. A virtual trainer could be used to start training end-users on the new system before it roles off of the production line.

4.5.7.5. M&S in Operations and Support

As systems are fielded end-user innovation and feedback on the operational performance of a system and its role in a FoS or SoS may necessitate design modifications. Operational maintenance and repairs can be compared to the projections made by the logistical models and simulations so that the models can be revalidated and modified. The end-user feedback can be incorporated into existing M&S tools used in the system's established collaborative environment to examine redesign alternatives. The operational and support phase can be considered the beginning of the acquisition cycle because this is when needed capabilities and new requirements are identified.

The M&S applied to the system's acquisition process has potential to be re-used as course-of-action, decision support, and training tools. Additionally, the program manager has an M&S repository that represents the system at multiple levels of fidelity that can be used to represent the system in other M&S FoS and SoS environments. Thereby, it is incumbent for a program manager to plan for maintaining the M&S used throughout the development of the system.

M&S plays an important role in all aspects of the acquisition process. This is especially true in designing and developing a capability that is part of a FoS or SoS. Today's systems and associated interactions are too complex and M&S can assist the process by controlling the desired variables to provide a repeatable audit trail that can assist in the acquisition decision processes.

4.5.7.6. M&S Resources

Properly implemented, M&S can ensure that schedules are met, costs and production constraints are identified and quantified, and system requirements and key performances are achieved. The following documents are provided for additional guidance. Additionally each service has a modeling and simulation office, which provides support to program offices.

Documents:

- DoD Directive 5000.59, Modeling and Simulation Management
- DoD 5000.59-M, Glossary of Modeling and Simulation Terms
- DoD 5000.59-P, Modeling and Simulation (M&S) Master Plan
- DoD Instruction 5000.61, Verification, Validation and Accreditation

Standards:

- IEEE 1278 (Series), IEEE Standard for Distributed Interactive Simulation (DIS)
- IEEE 1516 (Series), IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)

Websites:

- Defense Modeling & Simulation Office: www.dmso.mil
- Army Model and Simulation Office: www.amso.army.mil
- Navy Modeling and Simulation Management Office: www.navmsmo.hq.navy.mil
- Air Force Agency for Modeling and Simulation: www.afams.af.mil
- Simulation Interoperability Standards Organization: www.sisostds.org
- Institute of Electrical and Electronics Engineers: www.ieee.org

4.5.8. Summary of Technical Reviews

Technical reviews are an important oversight tool that the program manager can use to review and evaluate the state of the system and the program, re-directing activity after the review if found necessary. The commonly used reviews during most acquisition programs are the following:

- Initial Technical Review
- Alternative Systems Review
- System Requirements Review
- System Functional Review
- Preliminary Design Review
- Critical Design Review
- Test Readiness Review
- Production Readiness Review
- System Verification Review
- Operational Test Readiness Review

NOTE: The technical reviews listed above and described below are detailed reviews conducted between the program management office and contractor personnel to assist the program manager and contractor in assessing technical progress of the program. Unlike these technical reviews, a Design Readiness Review (DoD Instruction 5000.2) and Full-Rate Production Decision Review (DoD Instruction 5000.2) are Milestone Decision Authority-led management oversight reviews intended to provide an assessment (cost, schedule, and performance) of a program's readiness to progress further through the acquisition life cycle.

4.5.9. General Knowledge Tools

4.5.9.1. Best Practices

- The General Accounting Office has conducted several studies (A and B) on best practices
- The Systems Engineering Community of Practice
- The Systems Engineering Process Office within the Science, Technology, and Engineering Department of the Space and Naval Warfare Systems Center in San Diego, CA, is a resource for systems engineering and software engineering best practices. http://sepo.spawar.navy.mil/sepo/SEPOFlyer.html

4.5.9.2. Case Studies

- The Office of the Under Secretary of Defense (Acquisition, Technology, and Logistics), Office of Systems Engineering, has published several Integrated Product and Process Development case studies, including
  - Integrated Product/Process Development in the New Attack Submarine Program: A Case Study
  - Ford Motor Company's Investment Efficiency Initiative: A Case Study
  - Integrated Product/Process Development in Upgrade and Mod Programs.

- The Air Force Center for Systems Engineering has several case studies underway: C-5, F-111, Theater Battle Management Core System, and the Hubble Space Telescope. Case studies are also being planned for missile defense, DoD space-based systems, and commercial systems. http://cse.afit.edu/studies.htm
- Reliability, Availability and Maintainability Primer Case Studies

4.5.9.3. Lessons Learned

Lessons learned are a tool that the program manager may use to help identify potential areas of risk associated with the system by reviewing the experiences encountered in past programs. Lessons learned databases document what worked and what did not work in past programs, in the hopes that future programs can avoid the same pitfalls. Lessons learned can be found at all levels of the program, including: managerial, system, sub-system, and component.

Lessons learned are most effective when analogous programs and systems are identified, and the lessons learned are applied with discretion and proper judgment, as opposed to non-applicable lessons being blindly followed.

Ideally, a program manager searches lessons learned databases for analogous systems, enabling the program manager to be better prepared to defuse potential problems before they become real problems or to see what solutions to similar problems worked well in the past. However, because lessons learned databases are currently highly decentralized, it is often difficult to efficiently and effectively find applicable lessons learned in a form that is useful.

There are many organizations that produce lessons learned. Links to some of these organizations and databases from within and outside the DoD are given below.

- Center for Army Lessons Learned
- Air Force Center for Knowledge Sharing Lessons Learned
- Air Force Knowledge Management
- Navy Lessons Learned System
- Joint Center for Lessons Learned
- Department of Energy Lessons Learned
- NASA Lessons Learned Information System

4.6. Systems Engineering Resources

4.6.1. Standards and Models

- International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 15288, *System Life Cycle Processes*
- ISO/IEC 12207, *Software Life Cycle Processes*
- Electronic Industry Alliance (EIA)/Institute of Electrical and Electronic Engineers (IEEE) J-STD-016, *Software Development*
- American National Standards Institute (ANSI)/EIA 632, *Processes for Engineering a System*

- ANSI/EIA 649, *National Consensus Standard for Data Management*
- ANSI/EIA 748A, *Earned Value Management Systems*
- EIA 859, *Consensus Standard for Data Management*
- IEEE 1220, *Application Management of the Systems Engineering Process*
- EIA 731, *Systems Engineering Capability Model*
- CMMI SWE/SE/IPPD/SS, *Capability Maturity Model-Integration, Software Engineering, Systems Engineering, Integrated Product and Process Development and Supplier Sourcing*

4.6.2. Handbooks and Guides

- *Guidance for the Use of Robust Engineering in Air Force Acquisition Programs*
- *Navy Systems Engineering Guide*
- INCOSE Handbook
- MIL-HDB-61, *Configuration Management*
- MIL-HDBK 881, *Work Breakdown Structure*
- MIL-HDBK 1785, *Systems Security Engineering*
- *NASA SE Handbook*

  use the link below http://acc.dau.mil/simplify/file_download.php/182_NASA_SYS_ENGR_HDBK_SP-6105.pdf?URL_ID=7176&filename=103436167668182_NASA_SYS_ENGR_HDBK_SP-6105.pdf&filetype=application%2Fpdf&filesize=2644460&name=182_NASA_SYS_ENGR_HDBK_SP-6105.pdf&location=user-S/
- *DSMC Systems Engineering Fundamentals*
- *DAU Risk Management Handbook*
- *Product Support for the 21st Century: A Program Manager's Guide to Buying Performance*
- *Designing and Assessing Supportability in DoD Weapon Systems: A Guide to Increased Reliability and Reduced Logistics Footprint* use this link http://acc.dau.mil/simplify/file_download.php/FINAL+GUIDE+with+Memo+-+October+24.pdf?URL_ID=15943&filename=10772113271FINAL_GUIDE_with_Memo_-_October_24.pdf&filetype=application%2Fpdf&filesize=432407&name=FINAL+GUIDE+with+Memo+-+October+24.pdf&location=user-S/
- *DoD Template for Application of Total Life Cycle Systems Management (TLCSM) and Performance Based Logistics (PBL) In the Weapon System Life Cycle*
- *DoD Guide for Uniquely Identifying Items*
- The Reliability Analysis Center is a DoD Information Analysis Center, a Center of Excellence, and a technical focal point for information, data, analysis, training and

technical assistance in the engineering fields of Reliability, Maintainability, Supportability, and Quality. Their web site is http://rac.alionscience.com/

- ISO/IEC TR 19760, Systems Engineering – A guide for the application of ISO/IEC 15288 (System Life Cycle Processes), First Edition, 2003-11-15