ENGINEERING-PDH.com
ONLINE CONTINUING EDUCATION

FUNDAMENTALS OF SYSTEMS ENGINEERING PART 3

Main Category:	Project Management
Sub Category:	Systems Engineering
Course #:	PRJ-118
Course Content:	75 pgs
PDH/CE Hours:	5

OFFICIAL COURSE/EXAM

(SEE INSTRUCTIONS ON NEXT PAGE)

WWW.ENGINEERING-PDH.COM
TOLL FREE (US & CA): 1-833-ENGR-PDH (1-833-364-7734)
SUPPORT@ENGINEERING-PDH.COM

PRJ-118 EXAM PREVIEW

- TAKE EXAM! -

Instructions:

- At your convenience and own pace, review the course material below. When ready, click "Take Exam!" above to complete the live graded exam. (Note it may take a few seconds for the link to pull up the exam.) You will be able to re-take the exam as many times as needed to pass.
- Upon a satisfactory completion of the course exam, which is a score of 70% or better, you will be provided with your course completion certificate. Be sure to download and print your certificates to keep for your records.

Exam Preview:

1. According to the reference material, the event-based schedule, sometimes referred to as the Systems Engineering Master Schedule (SEMS) or Integrated Master Schedule (IMS) is a technical event-driven (not time-driven) plan primarily concerned with product and process development.
   a. True
   b. False
2. Which of the following options below is a type of product improvement strategies that is implemented after the deployment of a project?
   a. Design changes
   b. Upgrades
   c. Planned improvement
   d. Production modifications
3. According to the reference material, Product improvement planning must be driven by $ color or calendar, not by risk management.
   a. True
   b. False
4. Which of the following options below is NOT a type of Production Improvement Strategies for deployed systems upgrades?
   a. Major Rebuild
   b. Post-Production Improvement
   c. Block Upgrade
   d. Zoned Upgrades
5. Section 17.3 outlines the roles and responsibilities that both the government entity and the contractor have for a given contract. Which of the following below options is listed as a role/responsibility that the government has?
   a. Technical planning related to execution

    b. Designing and developing modifications
    c. Managing external interfaces
    d. Defining the new performance envelope
6. Which of the following options below is a system type that is outlined as an Open Systems Initiative according to the reference material?
    a. C4ISR
    b. P3I
    c. IPT
    d. E3
7. Which of the following options below is NOT a component listed in the Simplified Computer Resource Reference Model shown in Figure 17-6?
    a. Operating System
    b. Processor
    c. RAM
    d. Backplane
8. Which of the following options below is NOT a life cycle consideration for Interface Management according to the reference material?
    a. Time and Cost to upgrade a system is reduced.
    b. Open system approach enhances the use of competitive products to support the system
    c. Conformance management becomes a part of the life cycle configuration process
    d. Time to next iteration of system
9. According to the reference material, complex systems usually have stagnant configurations.
    a. True
    b. False
10. According to the reference material, when organizing for system development, the following order ranks the hierarchy of each group: System level Management team, system level design team, product/process teams.
    a. True
    b. False

PART 4

PLANNING, ORGANIZING, AND MANAGING

CHAPTER 1

INTRODUCTION TO SYSTEMS ENGINEERING MANAGEMENT

1.1 PURPOSE

The overall organization of this text is described in the Preface. This chapter establishes some of the basic premises that are expanded throughout the book. Basic terms explained in this chapter are the foundation for following definitions. Key systems engineering ideas and viewpoints are presented, starting with a definition of a system.

1.2 DEFINITIONS

A System Is …

Simply stated, a system is an integrated composite of people, products, and processes that provide a capability to satisfy a stated need or objective.

Systems Engineering Is…

Systems engineering consists of two significant disciplines: the technical knowledge domain in which the systems engineer operates, and systems engineering management. This book focuses on the process of systems engineering management.

Three commonly used definitions of systems engineering are provided by the best known technical standards that apply to this subject. They all have a common theme:

- A logical sequence of activities and decisions that transforms an operational need into a description of system performance parameters and a preferred system configuration. (MIL-STD-499A, *Engineering Management*, 1 May 1974. Now cancelled.)

- An interdisciplinary approach that encompasses the entire technical effort, and evolves into and verifies an integrated and life cycle balanced set of system people, products, and process solutions that satisfy customer needs. (EIA Standard IS-632, *Systems Engineering*, December 1994.)

- An interdisciplinary, collaborative approach that derives, evolves, and verifies a life-cycle balanced system solution which satisfies customer expectations and meets public acceptability. (IEEE P1220, *Standard for Application and Management of the Systems Engineering Process*, [Final Draft], 26 September 1994.)

In summary, systems engineering is an interdisciplinary engineering management process that evolves and verifies an integrated, life-cycle balanced set of system solutions that satisfy customer needs.

Systems Engineering Management Is…

As illustrated by Figure 1-1, systems engineering management is accomplished by integrating three major activities:

- Development phasing that controls the design process and provides baselines that coordinate design efforts,

- A systems engineering process that provides a structure for solving design problems and

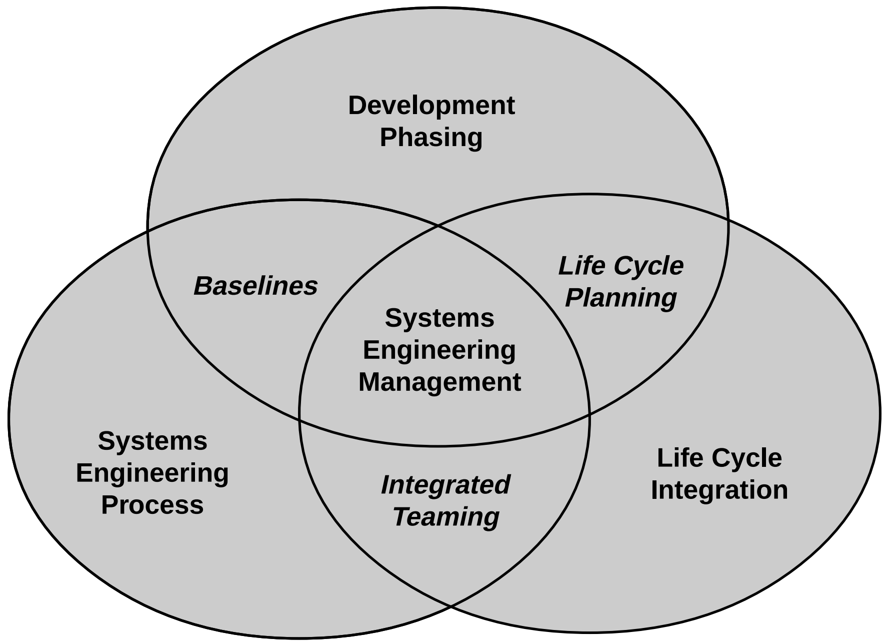


Figure 1-1. Three Activities of Systems Engineering Management

tracking requirements flow through the design effort, and

- Life cycle integration that involves customers in the design process and ensures that the system developed is viable throughout its life.

Each one of these activities is necessary to achieve proper management of a development effort. Phasing has two major purposes: it controls the design effort and is the major connection between the technical management effort and the overall acquisition effort. It controls the design effort by developing design baselines that govern each level of development. It interfaces with acquisition management by providing key events in the development process, where design viability can be assessed. The viability of the baselines developed is a major input for acquisition management Milestone (MS) decisions. As a result, the timing and coordination between technical development phasing and the acquisition schedule is critical to maintain a healthy acquisition program.

The systems engineering process is the heart of systems engineering management. Its purpose is to provide a structured but flexible process that transforms requirements into specifications, architectures, and configuration baselines. The discipline of this process provides the control and traceability to develop solutions that meet customer needs. The systems engineering process may be repeated one or more times during any phase of the development process.

Life cycle integration is necessary to ensure that the design solution is viable throughout the life of the system. It includes the planning associated with product and process development, as well as the integration of multiple functional concerns into the design and engineering process. In this manner, product cycle-times can be reduced, and the need for redesign and rework substantially reduced.

1.3 DEVELOPMENT PHASING

Development usually progresses through distinct levels or stages:

- Concept level, which produces a system concept description (usually described in a concept study);
- System level, which produces a system description in performance requirement terms; and
- Subsystem/Component level, which produces first a set of subsystem and component product performance descriptions, then a set of corresponding detailed descriptions of the products' characteristics, essential for their production.

The systems engineering process is applied to each level of system development, one level at a time, to produce these descriptions commonly called configuration baselines. This results in a series of configuration baselines, one at each development level. These baselines become more detailed with each level.

In the Department of Defense (DoD) the configuration baselines are called the functional baseline for the system-level description, the allocated baseline for the subsystem/ component performance descriptions, and the product baseline for the subsystem/component detail descriptions. Figure 1-2 shows the basic relationships between the baselines. The triangles represent baseline control decision points, and are usually referred to as technical reviews or audits.

Levels of Development Considerations

Significant development at any given level in the system hierarchy should not occur until the configuration baselines at the higher levels are considered complete, stable, and controlled. Reviews and audits are used to ensure that the baselines are ready for the next level of development. As will be shown in the next chapter, this review and audit process also provides the necessary assessment of system maturity, which supports the DoD Milestone decision process.

1.4 THE SYSTEMS ENGINEERING PROCESS

The systems engineering process is a top-down comprehensive, iterative and recursive problem

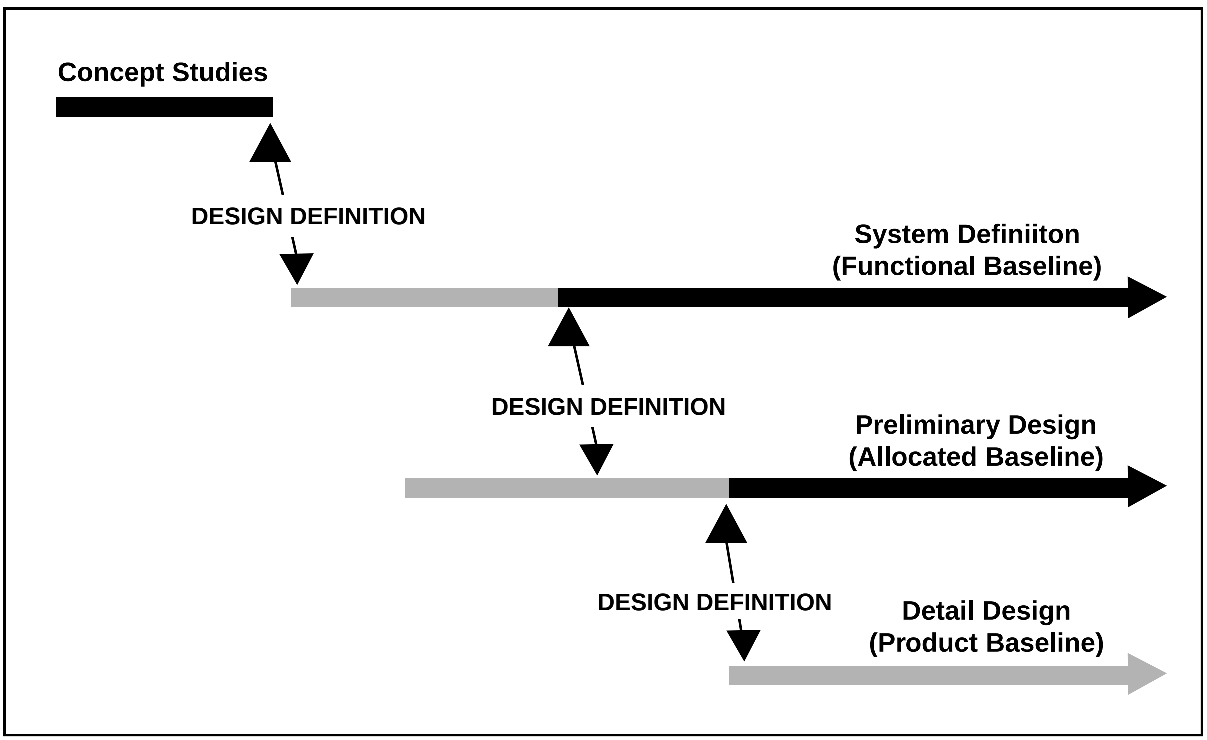


Figure 1-2. Development Phasing

solving process, applied sequentially through all stages of development, that is used to:

- Transform needs and requirements into a set of system product and process descriptions (adding value and more detail with each level of development),
- Generate information for decision makers, and
- Provide input for the next level of development.

As illustrated by Figure 1-3, the fundamental systems engineering activities are Requirements Analysis, Functional Analysis and Allocation, and Design Synthesis—all balanced by techniques and tools collectively called System Analysis and Control. Systems engineering controls are used to track decisions and requirements, maintain technical baselines, manage interfaces, manage risks, track cost and schedule, track technical performance, verify requirements are met, and review/audit the progress.

During the systems engineering process architectures are generated to better describe and understand the system. The word "architecture" is used in various contexts in the general field of engineering. It is used as a general description of how the subsystems join together to form the system. It can also be a detailed description of an aspect of a system: for example, the Operational, System, and Technical Architectures used in Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR), and software intensive developments. However, Systems Engineering Management as developed in DoD recognizes three universally usable architectures that describe important aspects of the system: functional, physical, and system architectures. This book will focus on these architectures as necessary components of the systems engineering process.

The *Functional Architecture* identifies and structures the allocated functional and performance requirements. The *Physical Architecture* depicts the

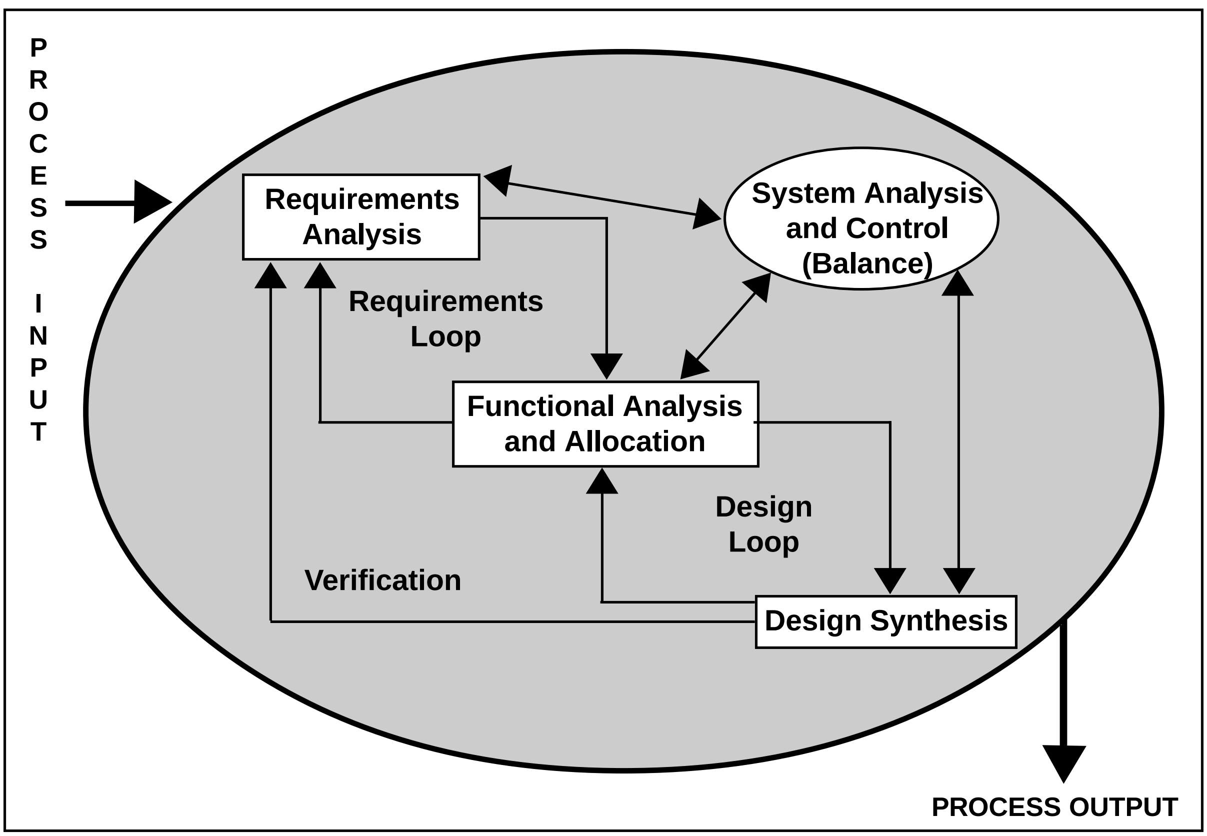


Figure 1-3. The Systems Engineering Process

system product by showing how it is broken down into subsystems and components. The *System Architecture* identifies all the products (including enabling products) that are necessary to support the system and, by implication, the processes necessary for development, production/construction, deployment, operations, support, disposal, training, and verification.

Life Cycle Integration

Life cycle integration is achieved through integrated development—that is, concurrent consideration of all life cycle needs during the development process. DoD policy requires integrated development, called Integrated Product and Product Development (IPPD) in DoD, to be practiced at all levels in the acquisition chain of command as will be explained in the chapter on IPPD. Concurrent consideration of all life cycle needs can be greatly enhanced through the use of interdisciplinary teams. These teams are often referred to as Integrated Product Teams (IPTs).

The objective of an Integrated Product Team is to:

- Produce a design solution that satisfies initially defined requirements, and
- Communicate that design solution clearly, effectively, and in a timely manner.

Multi-functional, integrated teams:

- Place balanced emphasis on product and process development, and
- Require early involvement of all disciplines appropriate to the team task.

Design-level IPT members are chosen to meet the team objectives and generally have distinctive competence in:

- Technical management (systems engineering),
- Life cycle functional areas (eight primary functions),
- Technical specialty areas, such as safety, risk management, quality, etc., or
- When appropriate, business areas such as finance, cost/budget analysis, and contracting.

Life Cycle Functions

Life cycle functions are the characteristic actions associated with the system life cycle. As illustrated by Figure 1-4, they are development, production and construction, deployment (fielding), operation, support, disposal, training, and verification. These activities cover the "cradle to grave" life cycle process and are associated with major functional groups that provide essential support to the life cycle process. These key life cycle functions are commonly referred to as the eight primary functions of systems engineering.

The customers of the systems engineer perform the life-cycle functions. The system user's needs are emphasized because their needs generate the requirement for the system, but it must be remembered that all of the life-cycle functional areas generate requirements for the systems engineering process once the user has established the basic need. ***Those that perform the primary functions also provide life-cycle representation in design-level integrated teams.***

Primary Function Definitions

Development includes the activities required to evolve the system from customer needs to product or process solutions.

Manufacturing/Production/Construction includes the fabrication of engineering test models and "brass boards," low rate initial production, full-rate production of systems and end items, or the construction of large or unique systems or subsystems.

Deployment (Fielding) includes the activities necessary to initially deliver, transport, receive, process, assemble, install, checkout, train, operate, house, store, or field the system to achieve full operational capability.

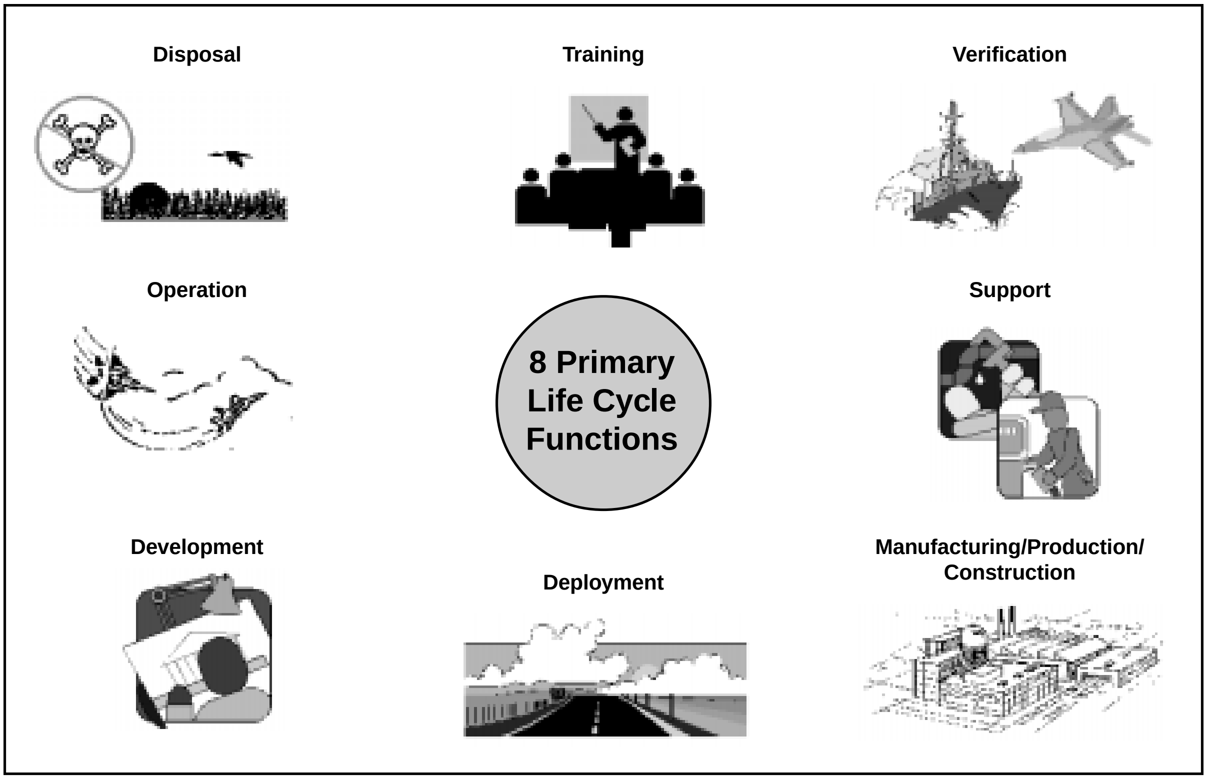


Figure 1-4. Primary Life Cycle Functions

Operation is the user function and includes activities necessary to satisfy defined operational objectives and tasks in peacetime and wartime environments.

Support includes the activities necessary to provide operations support, maintenance, logistics, and material management.

Disposal includes the activities necessary to ensure that the disposal of decommissioned, destroyed, or irreparable system components meets all applicable regulations and directives.

Training includes the activities necessary to achieve and maintain the knowledge and skill levels necessary to efficiently and effectively perform operations and support functions.

Verification includes the activities necessary to evaluate progress and effectiveness of evolving system products and processes, and to measure specification compliance.

Systems Engineering Considerations

Systems engineering is a standardized, disciplined management process for development of system solutions that provides a constant approach to system development in an environment of change and uncertainty. It also provides for simultaneous product and process development, as well as a common basis for communication.

Systems engineering ensures that the correct technical tasks get done during development through planning, tracking, and coordinating. Responsibilities of systems engineers include:

- Development of a total system design solution that balances cost, schedule, performance, and risk,
- Development and tracking of technical information needed for decision making,
- Verification that technical solutions satisfy customer requirements,

- Development of a system that can be produced economically and supported throughout the life cycle,
- Development and monitoring of internal and external interface compatibility of the system and subsystems using an open systems approach,
- Establishment of baselines and configuration control, and
- Proper focus and structure for system and major sub-system level design IPTs.

1.5 GUIDANCE

DoD 5000.2-R establishes two fundamental requirements for program management:

- It requires that an Integrated Product and Process approach be taken to design wherever practicable, and
- It requires that a disciplined systems engineering process be used to translate operational needs and/or requirements into a system solution.

Tailoring the Process

System engineering is applied during all acquisition and support phases for large- and small-scale systems, new developments or product improvements, and single and multiple procurements. The process must be tailored for different needs and/or requirements. Tailoring considerations include system size and complexity, level of system definition detail, scenarios and missions, constraints and requirements, technology base, major risk factors, and organizational best practices and strengths.

For example, systems engineering of software should follow the basic systems engineering approach as presented in this book. However, it must be tailored to accommodate the software development environment, and the unique progress tracking and verification problems software development entails. In a like manner, all technology domains are expected to bring their own unique needs to the process.

This book provides a conceptual-level description of systems engineering management. The specific techniques, nomenclature, and recommended methods are not meant to be prescriptive. Technical managers must tailor their systems engineering planning to meet their particular requirements and constraints, environment, technical domain, and schedule/budget situation.

However, the basic time-proven concepts inherent in the systems engineering approach must be retained to provide continuity and control. For complex system designs, a full and documented understanding of what the system must do should precede development of component performance descriptions, which should precede component detail descriptions. Though some parts of the system may be dictated as a constraint or interface, in general, solving the design problem should start with analyzing the requirements and determining what the system has to do before physical alternatives are chosen. Configurations must be controlled and risk must be managed.

Tailoring of this process has to be done carefully to avoid the introduction of substantial unseen risk and uncertainty. Without the control, coordination, and traceability of systems engineering, an environment of uncertainty results which will lead to surprises. Experience has shown that these surprises almost invariably lead to significant impacts to cost and schedule. Tailored processes that reflect the general conceptual approach of this book have been developed and adopted by professional societies, academia, industry associations, government agencies, and major companies.

1.6 SUMMARY POINTS

- Systems engineering management is a multifunctional process that integrates life cycle functions, the systems engineering problem-solving process, and progressive baselining.

- The systems engineering process is a problem-solving process that drives the balanced development of system products and processes.
- Integrated Product Teams should apply the systems engineering process to develop a life cycle balanced-design solution.
- The systems engineering process is applied to each level of development, one level at a time.
- Fundamental systems engineering activities are Requirements Analysis, Functional Analysis/Allocation, and Design Synthesis, all of which are balanced by System Analysis and Control.
- Baseline phasing provides for an increasing level of descriptive detail of the products and processes with each application of the systems engineering process.
- Baselining in a nut shell is a concept description that leads to a system definition which, in turn, leads to component definitions, and then to component designs, which finally lead to a product.
- The output of each application of the systems engineering process is a major input to the next process application.

CHAPTER 16

SYSTEMS ENGINEERING PLANNING

16.1 WHY ENGINEERING PLANS?

Systems engineering planning is an activity that has direct impact on acquisition planning decisions and establishes the feasible methods to achieve the acquisition objectives. Management uses it to:

- Assure that all technical activities are identified and managed,
- Communicate the technical approach to the broad development team,
- Document decisions and technical implementation, and
- Establish the criteria to judge how well the system development effort is meeting customer and management needs.

Systems engineering planning addresses the scope of the technical effort required to develop the system. The basic questions of "who will do what" and "when" are addressed. As a minimum, a technical plan describes what must be accomplished, how systems engineering will be done, how the effort will be scheduled, what resources are needed, and how the systems engineering effort will be monitored and controlled. The planning effort results in a management-oriented document covering the implementation of program requirements for system engineering, including technical management approaches for subsequent phases of the life cycle. In DoD it is an exercise done on a systems level by the government, and on a more detailed level by contractors.

Technical/Systems Engineering Planning

Technical planning may be documented in a separate engineering management plan or incorporated into a broad, integrated program management plan. This plan is first drafted at project or program inception during the early requirements analysis effort. Requirements analysis and technical planning are inherently linked, because requirements analysis establishes an understanding of what must be provided. This understanding is fundamental to the development of detailed plans.

To be of utility, systems engineering plans must be regularly updated. To support management decision making, major updates will usually occur at least just before major management milestone decisions. However, updates must be performed as necessary between management milestones to keep the plan sufficiently current to achieve its purpose of information, communication, and documentation.

16.2 ELEMENTS OF TECHNICAL PLANS

Technical plans should include sufficient information to document the purpose and method of the systems engineering effort. Plans should include the following:

- An introduction that states the purpose of the engineering effort and a description of the system being developed,
- A technical strategy description that ties the engineering effort to the higher-level management planning,

- A description of how the systems engineering process will be tailored and structured to complete the objectives stated in the strategy,
- An organization plan that describes the organizational structure that will achieve the engineering objectives, and
- A resource plan that identifies the estimated funding and schedule necessary to achieve the strategy.

Introduction

The introduction should include:

Scope: The scope of the plan should provide information concerning what part of the big picture the plan covers. For example, if the plan were a DoD program office plan, it would emphasize control of the higher-level requirements, the system definition (functional baseline), and all activities necessary for system development. On the other hand, a contractor's plan would emphasize control of lower-level requirements, preliminary and detail designs (allocated and product baselines), and activities required and limited by the contractual agreement.

Description: The description of the system should:

- Be limited to an executive summary describing those features that make the system unique,
- Include a general discussion of the system's operational functions, and
- Answer the question "What is it and what will it do?"

Focus: A guiding focus for the effort should be provided to clarify the management vision for the development approach. For example, the focus may be *lowest cost to obtain threshold requirements*, *superior performance within budget*, *superior standardization for reduced logistics*, *maximum use of the open systems approach to reduce cost*, or the like. A focus statement should:

- Be a single objective to avoid confusion,
- Be stated simply to avoid misinterpretation, and
- Have high-level support.

Purpose: The purpose of the engineering effort should be described in general terms of the outputs, both end products and life-cycle enabling products that are required. The stated purpose should answer the question, "What does the engineering effort have to produce?"

Technical Strategy

The basic purpose of a technical strategy is to link the development process with the acquisition or contract management process. It should include:

- Development phasing and associated baselining,
- Key engineering milestones to support risk management and business management milestones,
- Associated parallel developments or product improvement considerations, and
- Other management generated constraints or high-visibility activities that could affect the engineering development.

Phasing and Milestones: The development phasing and baseline section should describe the approach to phasing the engineering effort, including tailoring of the basic process described in this book and a rationale for the tailoring. The key milestones should be in general keeping with the technical review process, but tailored as appropriate to support business management milestones and the project/program's development phasing. Strategy considerations should also include discussion of how design and verification will phase into production and fielding. This area should identify how production will be phased-in (including use of limited-rate initial production and long lead-time purchases), and that initial support considerations require significant coordination between the user and acquisition community.

Parallel Developments and Product Improvement: Parallel development programs necessary for the system to achieve its objectives should be identified and the relationship between the efforts explained. Any product improvement strategies should also be identified. Considerations such as evolutionary development and preplanned product improvement should be described in sufficient detail to show how they would phase into the overall effort.

Impacts on Strategy

All conditions or constraints that impact the strategy should be identified and the impact assessed. Key points to consider are:

- Critical technologies development,
- Cost As an Independent Variable (CAIV), and
- Any business management directed constraint or activity that will have a significant influence on the strategy.

Critical Technologies: Discussion of critical technology should include:

- Risk associated with critical technology development and its impact on the strategy,
- Relationship to baseline development, and
- Potential impact on the overall development effort.

Cost As an Independent Variable: Strategy considerations should include discussion of how CAIV will be implemented, and how it will impact the strategy. It should discuss how unit cost, development cost, life cycle cost, total ownership cost, and their interrelationships apply to the system development. This area should focus on how these costs will be balanced, how they will be controlled, and what impact they have on the strategy and design approach.

Management Issues: Management issues that pose special concerns for the development strategy could cover a wide range of possible issues. In general, management issues identified as engineering strategy issues are those that impact the ability to support the management strategy. Examples would include:

- Need to combine developmental phases to accommodate management driven schedule or resource limitations,
- Risk associated with a tight schedule or limited budget,
- Contractual approach that increases technical risk, and
- Others of a similar nature.

Management-dictated technical activities—such as use of M&S, open systems, IPPD, and others—should not be included as a strategy issue unless they impact the overall systems engineering strategy to meet management expectations. The strategy discussion should lay out the plan, how it dovetails with the management strategy, and how management directives impact it.

Systems Engineering Processes

This area of the planning should focus on how the system engineering processes will be designed to support the strategy. It should include:

- Specific methods and techniques used to perform the steps and loops of the systems engineering process,
- Specific system analysis and control tools and how they will be used to support step and loop activities, and
- Special design considerations that must be integrated into the engineering effort.

Steps and Loops: The discussion of how the systems engineering process will be done should show the specific procedures and products that will ensure:

- Requirements are understood prior to the flow-down and allocation of requirements,
- Functional descriptions are established before designs are formulated,
- Designs are formulated that are traceable to requirements,
- Methods exist to reconsider previous steps, and
- Verification processes are in place to ensure that design solutions meet needs and requirements.

This planning area should address each step and loop for each development phase, include identification of the step-specific tools (Functional Flow Block Diagrams, Timeline Analysis, etc.) that will be used, and establish the verification approach. The verification discussion should identify all verification activities, the relationship to formal developmental T&E activities, and independent testing activities (such as operational testing).

Norms of the particular technical area and the engineering processes of the command, agency, or company doing the tasks will greatly influence this area of planning. However, whatever procedures, techniques, and analysis products or models used, they should be compatible with the basic principles of systems engineering management as described earlier in this book.

An example of the type of issue this area would address is the requirements analysis during the system definition phase. Requirements analysis is more critical and a more central focus during system definition than in later phases. The establishment of the correct set of customer requirements at the beginning of the development effort is essential to proper development. Accordingly, the system definition phase requirements analysis demands tight control and an early review to verify the requirements are established well enough to begin the design effort. This process of control and verification necessary for the system definition phase should be specifically described as part of the overall requirements analysis process and procedures.

Analysis and Control: Planning should identify those analysis tools that will be used to evaluate alternative approaches, analyze or assess effectiveness, and provide a rigorous quantitative basis for selecting performance, functional, and design requirements. These processes can include trade studies, market surveys, M&S, effectiveness analyses, design analyses, QFD, design of experiments, and others.

Planning must identify the method by which control and feedback will be established and maintained. The key to control is performance-based measurement guided by an event-based schedule. Entrance and exit criteria for the event-driven milestones should be established sufficient to demonstrate proper development progress has been completed. Event-based schedules and exit criteria are further discussed later in this chapter. Methods to maintain feedback and control are developed to monitor progress toward meeting the exit criteria. Common methods were discussed earlier in this book in the chapters on metrics, risk management, configuration management, and technical reviews.

Design Considerations: In every system development there are usually technical activities that require special attention. These may come from management concerns, legal or regulatory directives, social issues, or organizational initiatives. For example, a DoD program office will have to conform to DoDD 5000.2-R, which lists several technical activities that must be incorporated into the development effort. DoD plans should specifically address each issue presented in the Program Design section of DoD 5000.2-R.

In the case of a contractor there may be issues delineated in the contract, promised in the proposal, or established by management that the technical effort must address. The system engineering planning must describe how each of these issues will be integrated into the development effort.

Organization

Systems engineering management planning should identify the basic structure that will develop the system. Organizational planning should address how the integration of the different technical disciplines, primary function managers, and other stakeholders will be achieved to develop the system. This planning area should describe how multidisciplinary teaming would be implemented, that is, how the teams will be organized, tasked, and trained. A systems-level team should be established early to support this effort. Roles, authority, and basic responsibilities of the system-level design team should be specifically described. Establishing the design organization should be one of the initial tasks of the system-level design team. Their basic approach to organizing the effort should be described in the plan. Further information on organizing is contained in a later chapter.

Resources

The plan should identify the budget for the technical development. The funds required should be matrixed against a calendar schedule based on the event-based schedule and the strategy. This should establish the basic development timeline with an associated high-level estimated spending profile. Shortfalls in funding or schedule should be addressed and resolved by increasing funds, extending schedule, or reducing requirements prior to the plan preparation. Remember that future analysis of development progress by management will tend to be based on this budget "promised" at plan inception.

16.3 INTEGRATION OF PLANS – PROGRAM PLAN INTERFACES

Systems engineering management planning must be coordinated with interfacing activities such as these:

- Acquisition Strategy assures that technical plans take into account decisions reflected in the Acquisition Strategy. Conflicts must be identified early and resolved.
- Financial plan assures resources match the needs in the tech plan. Conflicts should be identified early and resolved.
- Test and Evaluation Master Plan (TEMP) assures it complements the verification approach. It should provide an integrated approach to verify that the design configuration will meet customer requirements. This approach should be compatible with the verification approach delineated in the systems engineering plan.
- Configuration management plan assures that the development process will maintain the system baselines and control changes to them.
- Design plans (e.g., electrical, mechanical, structural, etc.) coordinates identification of IPT team composition.
- Integrated logistics support planning and support analysis coordinates total system support.
- Production/Manufacturing plan to coordinate activities concerning design producibility, and follow-on production,
- Quality management planning assures that quality engineering activities and quality management functions are included in system engineering planning,
- Risk management planning establishes and coordinates technical risk management to support total program risk management.
- Interoperability planning assures interoperability suitability issues are coordinated with system engineering planning. (Where interoperability is an especially critical requirement such as, communication or information systems, it should be addressed as a separate issue with separate integrated teams, monitoring, and controls).
- Others such as M&S plan, software development plan, human integration plan, environment, safety and health planning, also interface.

Things to Watch

A well developed technical management plan will include:

- The expected benefit to the user,
- How a total systems development will be achieved using a systems engineering approach,
- How the technical plan complements and supports the acquisition or management business plan,
- How incremental reviews will assure that the development stays on track,
- How costs will be reduced and controlled,
- What technical activities are required and who will perform them,
- How the technical activities relate to work accomplishment and calendar dates,
- How system configuration and risk will be controlled,
- How system integration will be achieved,
- How the concerns of the eight primary life cycle functions will be satisfied,
- How regulatory and contractual requirements will be achieved, and
- The feasibility of the plan, i.e., is the plan practical and executable from a technical, schedule, and cost perspective.

16.4 SUMMARY POINTS

- Systems engineering planning should establish the organizational structure that will achieve the engineering objectives.
- Planning must include event-based scheduling and establish feedback and control methods.
- It should result in important planning and control documents for carrying out the engineering effort.
- It should identify the estimated funding and detail schedule necessary to achieve the strategy.
- Systems engineering planning should establish the proper relationship between the acquisition and technical processes.

APPENDIX 16-A

SCHEDULES

The event-based schedule, sometimes referred to as the Systems Engineering Master Schedule (SEMS) or Integrated Master Schedule (IMS) is a technical event-driven (not time-driven) plan primarily concerned with product and process development. It forms the basis for schedule control and progress measurement, and relates engineering management events and accomplishments to the WBS. These events are identified either in the format of entry and exit events (e.g. initiate PDR, complete PDR) or by using entry and exit criteria for each event. Example exit criteria shown in Figures 16-1 and 16-2.

The program office develops an event-based schedule that represents the overall development effort. This schedule is usually high-level and focused on the completion of events that support the acquisition milestone decision process. An event-based schedule is developed by the contractor to include significant accomplishments that must be completed in order to meet the progress required prior to contract established events. The contractor also includes events, accomplishments, and associated success criteria specifically identified by the contract. DoD program offices can use the contractor's event-based schedule and the

System Requirements Review (SRR)	System Functional Review/Software Spec Review(SFR/SSR)	Preliminary Design Review (PDR)
• Mission Analysis completed • Support Strategy defined • System options decisions completed • Design usage defined • Operational performance requirement defined • Manpower sensitivities completed • Operational architecture available and reviewed	• Installed environments defined • Maintenance concept defined • Preliminary design criteria established • Preliminary design margins established • Interfaces defined/preliminary interface specs completed • Software and software support requirements completed • Baseline support/resources requirements defined • Support equipment capability defined • Technical architecture prepared • System defined and requirements shown to be achievable	• Design analyses/definition completed • Material/parts characterization completed • Design maintainability analysis completed/support requirements defined • Preliminary production plan completed • Make/buy decisions finalized • Breadboard investigations completed • Coupon testing completed • Design margins completed • Preliminary FMECA completed • Software functions and architecture and support defined • Maintenance tasks trade studies completed • Support equipment development specs completed

Figure 16-1. Sample Event-Based Schedule Exit Criteria

Critical Design Review Test Readiness Review (CDR/TRR)	System Verfication Review/ Functional Configuration Audit (SVR/FCA)	Physical Configuration Audit (PCA)
• Parts, materials, processes selected • Development tests completed • Inspection points/criteria completed • Component level FMECA completed • Repair level analysis completed • Facility requirements defined • Software test descriptions completed • Hardware and software hazard analysis completed • Firmware spt completed • Software programmers manual completed • Durability test completed • Maintainability analyses completed • Qualification test procedures approved • Producibility analyses completed	• All verification tasks completed • Durability tests completed • Long lead time items identified • PME and operational training completed • Tech manuals completed • Flight test plan approved • Support and training equipment developed • Fielding analysis completed • Provisioning data verified	• Qualification testing completed • All QA provisions finalized • All manufacturing process requirements and documentation finalized • Product fabrication specifications finalized • Support and training equipment qualification completed • All acceptance test requirements completed • Life management plan completed • System support capability demonstrated • Post production support analysis completed • Final software description document and all user manuals complete

Figure 16-2. Sample Event-Driven Schedule Exit Criteria (continued)

contractor's conformance to it for several purposes: source selection, monitoring contractor progress, technical and other reviews, readiness for option award, incentives/awards determination, progress payments decision, and similar activities.

The event-based schedule establishes the key parameters for determining the progress of a development program. To some extent it controls and interfaces with systems engineering management planning, integrated master schedules and integrated master plans, as well as risk management planning, system test planning, and other key plans which govern the details of program management.

The calendar or detail schedule is a time-based schedule that shows how work efforts will support tasks and events identified in the event-based schedule. It aligns the tasks and calendar dates to show when each significant accomplishment must be achieved. It is a key component for developing Earned Value metrics. The calendar schedule is commonly referred to as the detail schedule, systems engineering detail schedule, or SEDS. The contractor is usually required to maintain the relationship between the event and calendar schedules for contract required activities. Figure 16-3 shows the relationship between the system requirements, the WBS, the contractual requirements, the event-based schedule, and the detail schedule.

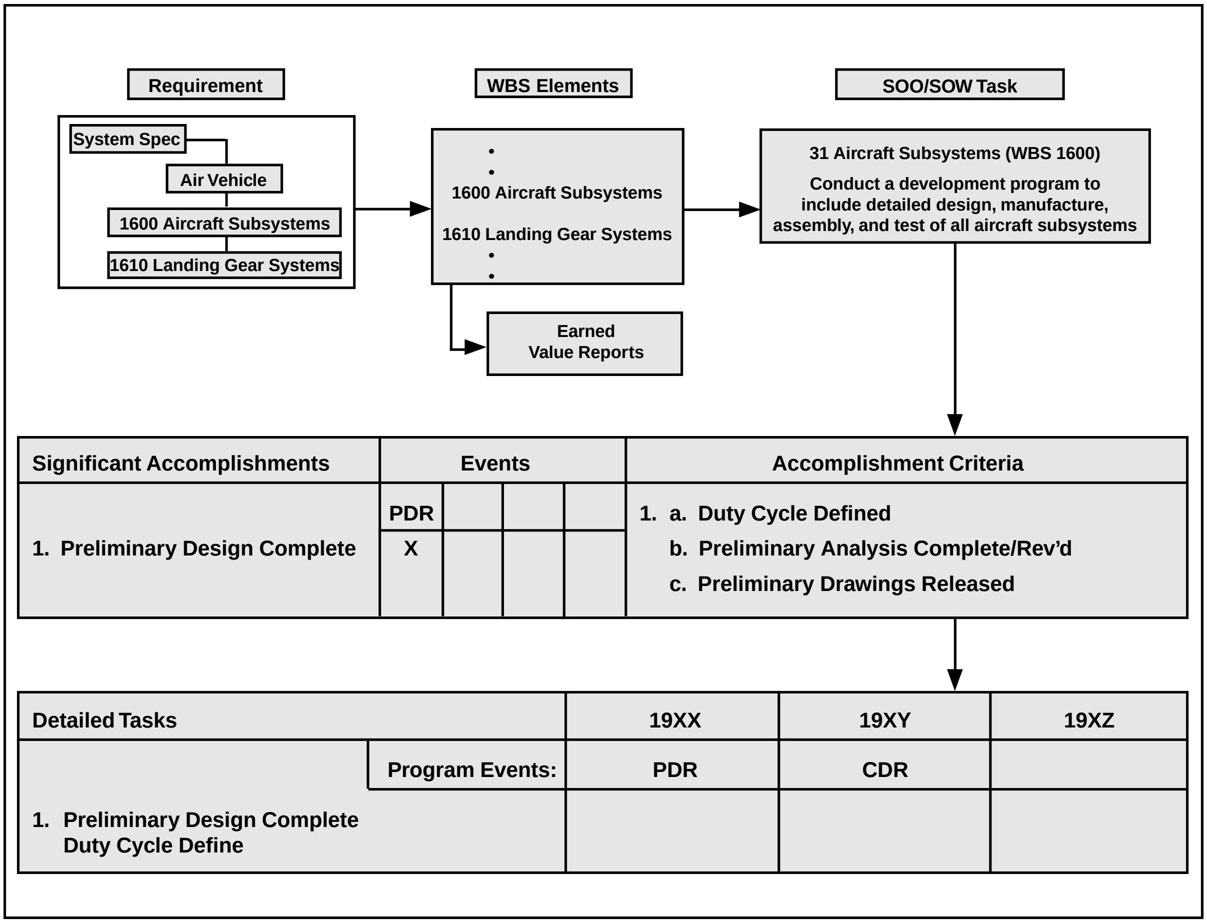


Figure 16-3. Event-Based—Detailed Schedule Interrelationships

Schedule Summary

The event-based schedule establishes the key tasks and results expected. The event-based schedule establishes the basis for a valid calendar-based (detail) schedule.

CHAPTER 17

PRODUCT IMPROVEMENT STRATEGIES

17.1 INTRODUCTION

Complex systems do not usually have stagnant configurations. A need for a change during a system's life cycle can come from many sources and effect the configuration in infinite ways. The problem with these changes is that, in most cases it is difficult, if not impossible, to predict the nature and timing of these changes at the beginning of system development. Accordingly, strategies or design approaches have been developed to reduce the risk associated with predicted and unknown changes.

Well thought-out improvement strategies can help control difficult engineering problems related to:

- Requirements that are not completely understood at program start,
- Technology development that will take longer than the majority of the system development,
- Customer needs (such as the need to combat a new military threat) that have increased, been upgraded, are different, or are in flux,
- Requirements change due to modified policy, operational philosophy, logistics support philosophy, or other planning or practices from the eight primary life cycle function groups,
- Technology availability that allows the system to perform better and/or less expensively,
- Potential reliability and maintainability upgrades that make it less expensive to use, maintain, or support, including development of new supply support sources,
- Safety issues requiring replacement of unsafe components, and
- Service life extension programs that refurbish and upgrade systems to increase their service life.

In DoD, the 21st century challenge will be improving existing products and designing new ones that can be easily improved. With the average service life of a weapons system in the area of 40 or more years, it is necessary that systems be developed with an appreciation for future requirements, foreseen and unforeseen. These future requirements will present themselves as needed upgrades to safety, performance, supportability, interface compatibility, or interoperability; changes to reduce cost of ownership; or major rebuild. Providing these needed improvements or corrections form the majority of the systems engineer's post-production activities.

17.2 PRODUCT IMPROVEMENT STRATEGIES

As shown by Figure 17-1, these strategies vary based on where in the life cycle they are applied. The strategies or design approaches that reflect these improvement needs can be categorized as planned improvements, changes in design or production, and deployed system upgrades.

Planned Improvements

Planned improvements strategies include evolutionary acquisition, preplanned product development, and open systems. These strategies are not exclusive and can be combined synergistically in a program development.

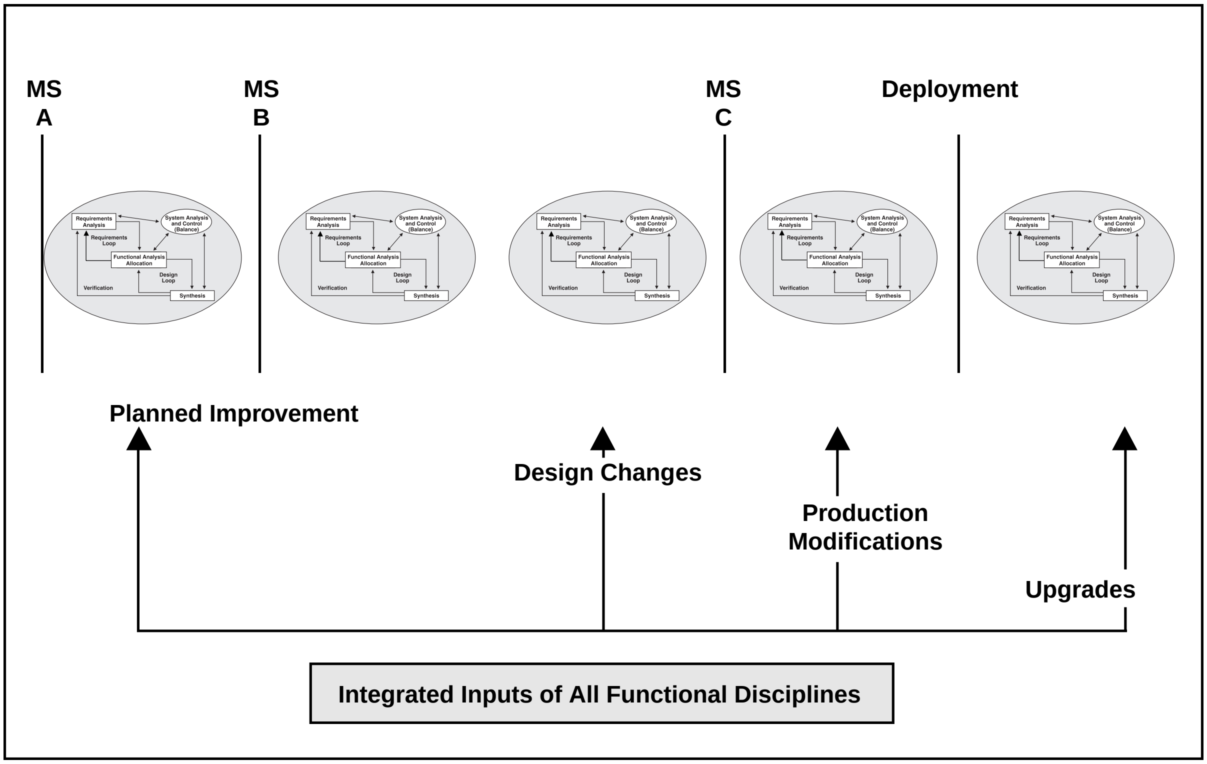


Figure 17-1. Types of Product Improvement Strategies

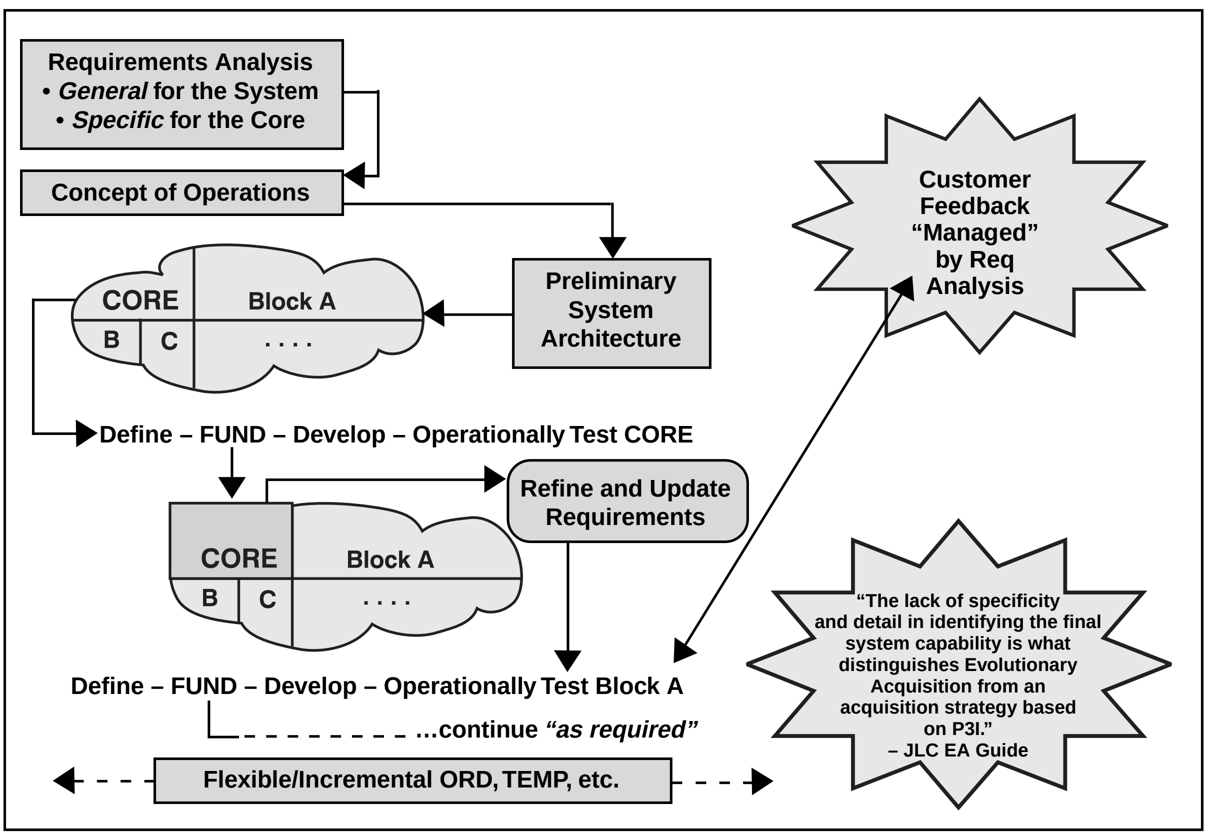


Figure 17-2. Evolutionary Acquisition

Evolutionary Acquisition: Evolutionary acquisition is the preferred approach to systems acquisition in DoD. In an environment where technology is a fast moving target and the key to military superiority is a technically superior force, the requirement is to transition useful capability from development to the user as quickly as possible, while laying the foundation for further changes to occur at later dates. Evolutionary acquisition is an approach that defines requirements for a core capability, with the understanding that the core is to be augmented and built upon (evolved) until the system meets the full spectrum of user requirements. The core capability is defined as a function of user need, technology maturity, threat, and budget. The core is then expanded as need evolves and the other factors mentioned permit.

A key to achieving evolutionary acquisition is the use of time-phased requirements and continuous communication with the eventual user, so that requirements are staged to be satisfied incrementally, rather than in the traditional single grand design approach. Planning for evolutionary acquisition also demands that engineering designs be based on open system, modular design concepts that permit additional increments to be added over time without having to completely re-design and re-develop those portions of the system already fielded. Open designs will facilitate access to recent changes in technologies and will also assist in controlling costs by taking advantage of commercial competition in the marketplace. This concept is not new; it has been employed for years in the C4ISR community, where system are often in evolution over the entire span of their lifecycles.

Preplanned Product Improvement (P3I): Often referred to as P3I, preplanned product improvement is an appropriate strategy when requirements are known and firm, but where constraints (typically either technology or budget) make some portion of the system unachievable within the schedule required. If it is concluded that a militarily

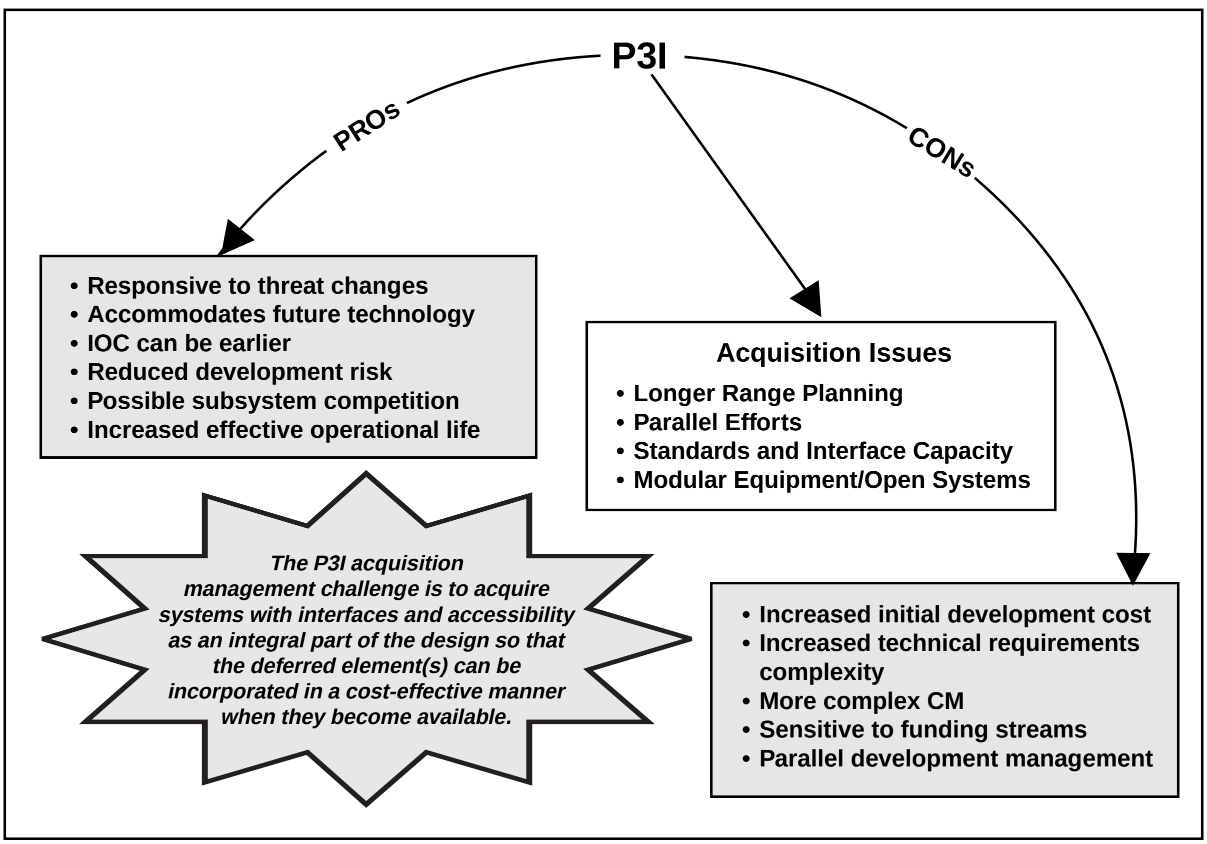


Figure 17-3. Pre-Planned Product Improvement

useful capability can be fielded as an interim solution while the portion yet to be proceeds through development, then P3I is appropriate. The approach generally is to handle the improvement as a separate, parallel development; initially test and deliver the system without the improvement; and prove and provide the enhanced capability as it becomes available. The key to a successful P3I is the establishment of well-defined interface requirements for the system and the improvement. Use of a P3I will tend to increase initial cost, configuration management activity, and technical complexity. Figure 17-3 shows some of the considerations in deciding when it is appropriate.

Open Systems Approach: The open system design approach uses interface management to build flexible design interfaces that accommodate use of competitive commercial products and provide enhanced capacity for future change. It can be used to prepare for future needs when technology is yet not available, whether the operational need is known or unknown. The open systems focus is to design the system such that it is easy to modify using standard interfaces, modularity, recognized interface standards, standard components with recognized common interfaces, commercial and nondevelopmental items, and compartmentalized design. Open system approaches to design are further discussed at the end of this chapter.

Changes in Design or Production

Engineering Change Proposals (ECPs): Changes that are to be implemented during the development and production of a given system are typically initiated through the use of ECPs. If the proposed change is approved (usually by a configuration control board) the changes to the documentation that describes the system are handled by formal configuration management, since, by definition, ECPs, when approved, change an approved baseline. ECPs govern the scope and details of these changes. ECPs may address a variety of needs, including correction of deficiencies, cost reduction, and safety. Furthermore, ECPs may been assigned differing levels of priority from routine to emergency. MIL-HDBK-61, Configuration Management Guidance, offers an excellent source of advice on issues related to configuration changes.

Block Change before Deployment: Block changes represent an attempt to improve configuration management by having a number of changes grouped and applied such that they will apply consistently to groups (or blocks) of production items. This improves the management and configuration control of similar items substantially in comparison to change that is implemented item by item and single change order by single change order. When block changes occur, the life cycle impact should be carefully addressed. Significant differences in block configurations can lead to different manuals, supply documentation, training, and restrictions as to locations or activities where the system can be assigned.

Deployed Systems Upgrades

Major Rebuild: A major rebuild results from the need for a system that satisfies requirements significantly different or increased from the existing system, or a need to extend the life of a system that is reaching the end of its usable life. In both cases the system will have upgraded requirements and should be treated as basically a new system development. A new development process should be started to establish and control configuration baselines for the rebuilt system based on the updated requirements.

Major rebuilds include remanufacturing, service-life extension programs, and system developments where significant parts of a previous system will be reused. Though rebuilding existing systems can dramatically reduce the cost of a new system in some cases, the economies of rebuild can be deceiving, and the choice of whether to pursue a rebuild should be done after careful use of trade studies. The key to engineering such systems is to remember that they are new systems and require the full developmental considerations of baselining, the systems engineering process, and life cycle integration.

Post-Production Improvement: In general, product improvements become necessary to improve the system or to maintain the system as its components reach obsolescence. These projects generally result in a capability improvement, but for all practical purposes the system still the serves the same basic need. These improvements are usually characterized by an upgrade to a component or subsystem as opposed to a total system upgrade.

Block Upgrades: Post-production block upgrades are improvements to a specific group of the system population that provides a consistent configuration within that group. Block upgrades in post-production serve the same general purpose of controlling individual system configurations as production block upgrades, and they require the same level of life-cycle integration.

Modifying an Existing System

Upgrading an existing system is a matter of following the system engineering process, with an emphasis on configuration and interface management. The following activities should be included when upgrading a system:

- Benchmark the modified requirements both for the upgrade and the system as a whole,
- Perform functional analysis and allocation on the modified requirements,
- Assess the actual capability of the pre-upgrade system,
- Identify cost and risk factors and monitor them,
- Develop and evaluate modified system alternatives,
- Prototype the chosen improvement alternative, and
- Verify the improvement.

Product improvement requires special attention to configuration and interface management. It is not uncommon that the existing system's configuration will not be consistent with the existing configuration data. Form, fit, and especially function interfaces often represent design constraints that are not always readily apparent at the outset of a system upgrade. Upgrade planning should ensure that the revised components will be compatible at the interfaces. Where interfaces are impacted, broad coordination and agreement is normally required.

Traps in Upgrading Deployed Systems

When upgrading a deployed system pay attention to the following significant traps:

Scheduling to minimize operational impacts: The user's operational commitments will dictate the availability of the system for modification. If the schedule conflicts with an existing or emerging operational need, the system will probably not become available for modification at the time agreed to. Planning and contractual arrangements must be flexible enough to accept unforeseen schedule changes to accommodate user's unanticipated needs.

Configuration and interface management: Configuration management must address three configurations: the actual existing configuration, the modification configuration, and the final system configuration. The key to successful modification is the level of understanding and control associated with the interfaces.

Logistics compatibility problems: Modification will change the configuration, which in most cases will change the supply support and maintenance considerations. Coordination with the logistics community is essential to the long-term operational success of the modification.

Minimal resources available: Modifications tend to be viewed as simple changes. As this chapter has pointed out, they are not; and they should be carefully planned. That planning should include an estimate of needed resources. If the resources are not available, either the project should be abandoned, or a plan formulated to mitigate and control the risk of an initial, minimal budget combined with a plan for obtaining additional resources.

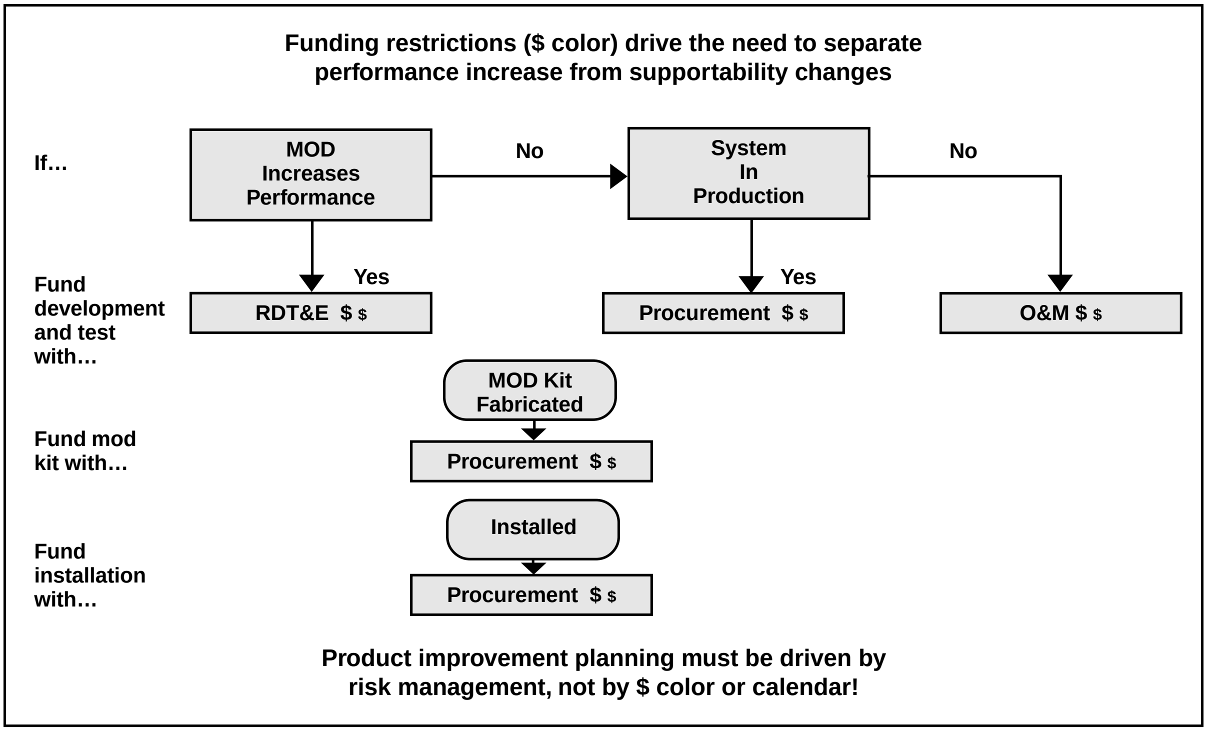


Figure 17-4. Funding Rule for DoD System Upgrades

Limited competitors: Older systems may have only a few suppliers that have a corporate knowledge of the particular system functions and design. This is especially problematic if the original system components were commercial or NDIs that the designer does not have product baseline data for. In cases such as these, there is a learning process that must take place before the designer or vendor can adequately support the modification effort. Depending on the specific system, this could be a major effort. This issue should be considered very early in the modification process because it has serious cost implications.

Government funding rules: As Figure 17-4 shows the use of government funding to perform system upgrades has restrictions. The purpose of the upgrade must be clear and justified in the planning efforts.

17.3 ROLES AND RESPONSIBILITIES

Modification management is normally a joint government and contractor responsibility. Though any specific system upgrade will have relationships established by the conditions surrounding the particular program, government responsibilities would usually include:

- Providing a clear statement of system requirements,
- Planning related to government functions,
- Managing external interfaces,
- Managing the functional baseline configuration, and
- Verifying that requirements are satisfied.

Contractor responsibilities are established by the contract, but would normally include:

- Technical planning related to execution,
- Defining the new performance envelope,
- Designing and developing modifications, and

- Providing evidence that changes made have modified the system as required.

System Engineering Role

The systems engineering role in product improvement includes:

- Planning for system change,
- Applying the systems engineering process,
- Managing interface changes,
- Identifying and using interface standards which facilitate continuing change,
- Ensuring life cycle management is implemented,
- Monitoring the need for system modifications, and
- Ensuring operations, support activities, and early field results are considered in planning.

17.4 SUMMARY POINTS

- Complex systems do not usually have stagnant configurations.
- Planned improvements strategies include evolutionary acquisition, preplanned product development, and open systems.
- A major rebuild should be treated as a new system development.
- Upgrading an existing system is a matter of following the system engineering process, with an emphasis on configuration and interface management.
- Pay attention to the traps. Upgrade projects have many.

SUPPLEMENT 17-A

OPEN SYSTEM APPROACH

The open system approach is a business and technical approach to system development that results in systems that are easier to change or upgrade by component replacement. It is a system development logic that emphasizes flexible interfaces and maximum interoperability, optimum use of commercial competitive products, and enhanced system capacity for future upgrade. The value of this approach is that open systems have flexibility, and that flexibility translates into benefits that can be recognized from business, management, and technical perspectives.

From a management and business view, the open system approach directs resources to a more intensive design effort with the expectation of a life cycle cost reduction. As a business approach it supports the DoD policy initiatives of CAIV, increased competition, and use of commercial products. It is a technical approach that emphasizes systems engineering, interface control, modular design, and design for upgrade. As a technical approach it supports the engineering goals of design flexibility, risk reduction, configuration control, long-term supportability, and enhanced utility.

Open Systems Initiative

In DoD the open system initiative was begun as a result of dramatic changes in the computer industry that afforded significant advantages to design of C4ISR and IT systems. The standardization achieved by the computer industry allows C4ISR and IT systems to be designed using interface standards to select off-the-shelf components to form the system. This is achieved by using commercially-supported specifications and standards for specifying system interfaces (external and internal, functional and physical), products, practices, and tools. An open system is one

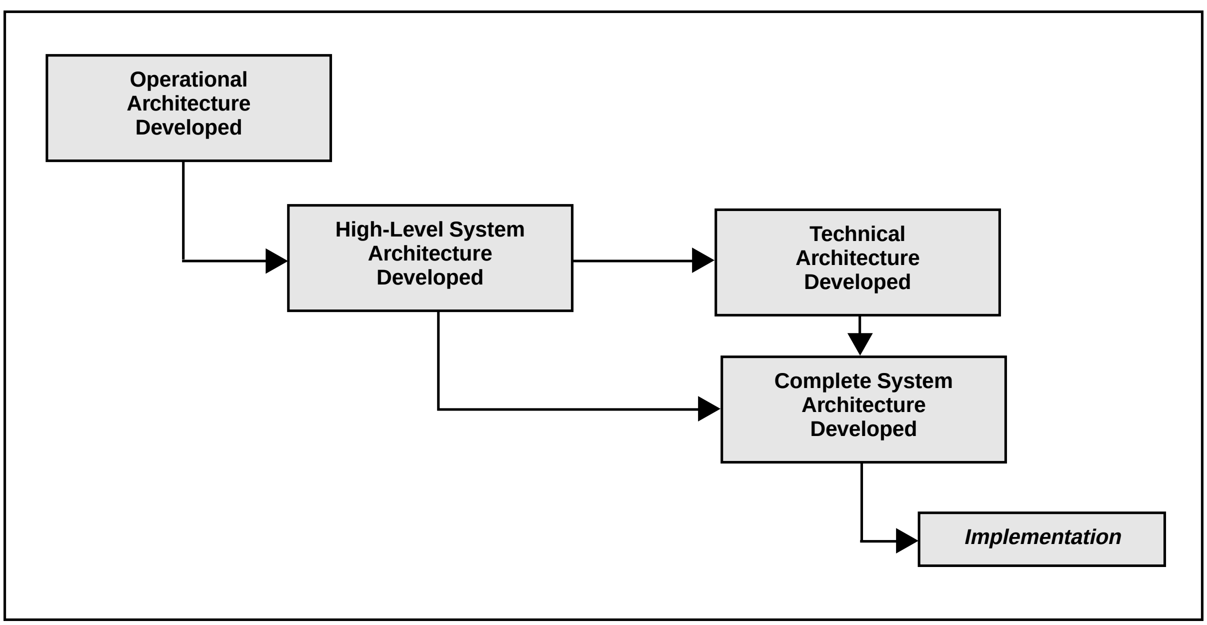


Figure 17-5. C4I and IT Development

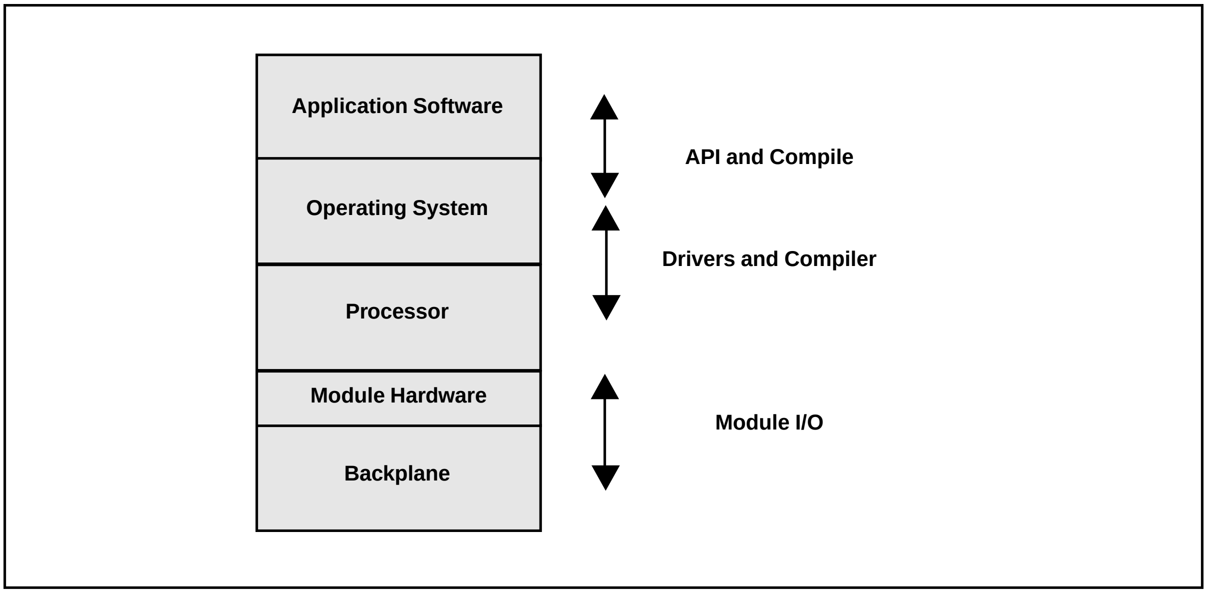


Figure 17-6. Simplified Computer Resource Reference Model

in which interfaces are fully described by open standards.[1] An open system approach extends this concept further by using modular design and interface design to enhance the availability of multiple design solutions, especially those reflecting use of open standards, competitive commercial components, NDIs, and future upgrade capability.

As developed in the C4ISR and IT communities, the open system approach requires the design of three architectures: operational, technical, and system.

As shown in Figure 17-5, the first one prepared is an operational architecture that defines the tasks, operational elements, and information flows required to accomplish or support an operational function. The user community generates the operational concepts that form an operational architecture. The operational architecture is allusive. It is not a specific document required to be developed by the user such as the ORD; but because of their operational nature, the user must provide the components of the operational architecture. It is usually left to the developer to assemble and structure the information as part of the system definition requirements analysis. Once the operational architecture has clearly defined the operational need, development of a system architecture[2] is begun.

The (open) system architecture is a set of descriptions, including graphics, of systems and interconnections supporting the operational functions described in the operational architecture. Early in the (open) system architecture development a technical architecture is prepared to establish a set of rules, derived from open consensus-based industry standards, to govern the arrangement, interaction, and interdependence of the elements of a reference model. Reference models are a common conceptual framework for the type of system being designed. (A simple version for computer resources is shown in Figure 17-6.)

[1] Open Standards are non-proprietary, consensus-based standards widely accepted by industry. Examples include SAE, IEEE, and ISO standards.

[2] This system architecture typically describes the end product but not the enabling products. It relies heavily on interface definitions to describe system components.

The technical architecture identifies the services, interfaces, standards, and their relationships; and provides the technical guidelines upon which engineering specifications are based, common building blocks are built, and product lines are developed. In short, the technical architecture becomes a design requirement for developing the system. (The purpose, form, and function of the technical architecture is similar to building codes.)

The system architecture is then further developed to eventually specify component performance and interface requirements. These are then used to select the specific commercial components that form the system under development. This process, called an *implementation*, envisions the production process as consisting primarily of selecting components, conformance (to the interface and performance requirements) management, and assembly, with little or no need for detailed design fabrications.

The process described above has allowed significant achievements in computer-related developments. Other technical fields have also used the open system design approach extensively. (Common examples are the electrical outlets in your home and the tire-to-wheel interface on your car). In most cases the process is not as well defined as it is in the current digital electronics area. A consistent successful use of the open design concept, in and outside the electronics field, requires an understanding of how this process relates to the activities associated with systems engineering management.

Systems Engineering Management

The open system approach impacts all three essential elements of systems engineering management: systems engineering phasing, the systems engineering process, and life cycle considerations. It requires enhanced interface management in the systems engineering process, and requires specific design products be developed prior to engineering-event milestones. The open systems approach is inherently life-cycle friendly. It favorably impacts production and support functions, but it also requires additional effort to assure life-cycle conformance to interface requirements.

Open Systems Products and SE Development Phasing

A system is developed with stepped phases that allow an understanding of the operational need to eventually evolve into a design solution. Though some tailoring of this concept is appropriate, the basic phasing (based on the operational concept preceding the system description, which precedes the preliminary design, which precedes the detailed design) is necessary to coordinate the overall design process and control the requirements flowdown. As shown by Figure 17-7 the open system approach blends well with these development phases.

Concept Studies Phase

The initial detailed operational concept, including operational architectures, should be a user-community output (with some acquisition engineering assistance) produced during the concept exploration phase that emphasizes operational concepts associated with various material solutions. The operational concept is then updated as necessary for each following phase. Analysis of the initial operational concept should be a key element of the operational view output of the system definition phase requirements analysis. An operational architecture developed for supporting the system description should be complete, comprehensive, and clear; and verified to be so at the Alternative Systems Review. If the operational architecture cannot be completed, then a core operational capability must be developed to establish the basis for further development. Where a core capability is used, core requirements should be complete and firm, and the process for adding expanded requirements should be clear and controlled.

System Definition Phase

System interface definitions, such as the technical architecture, and high-level (open) system architecture should be complete in initial form at the

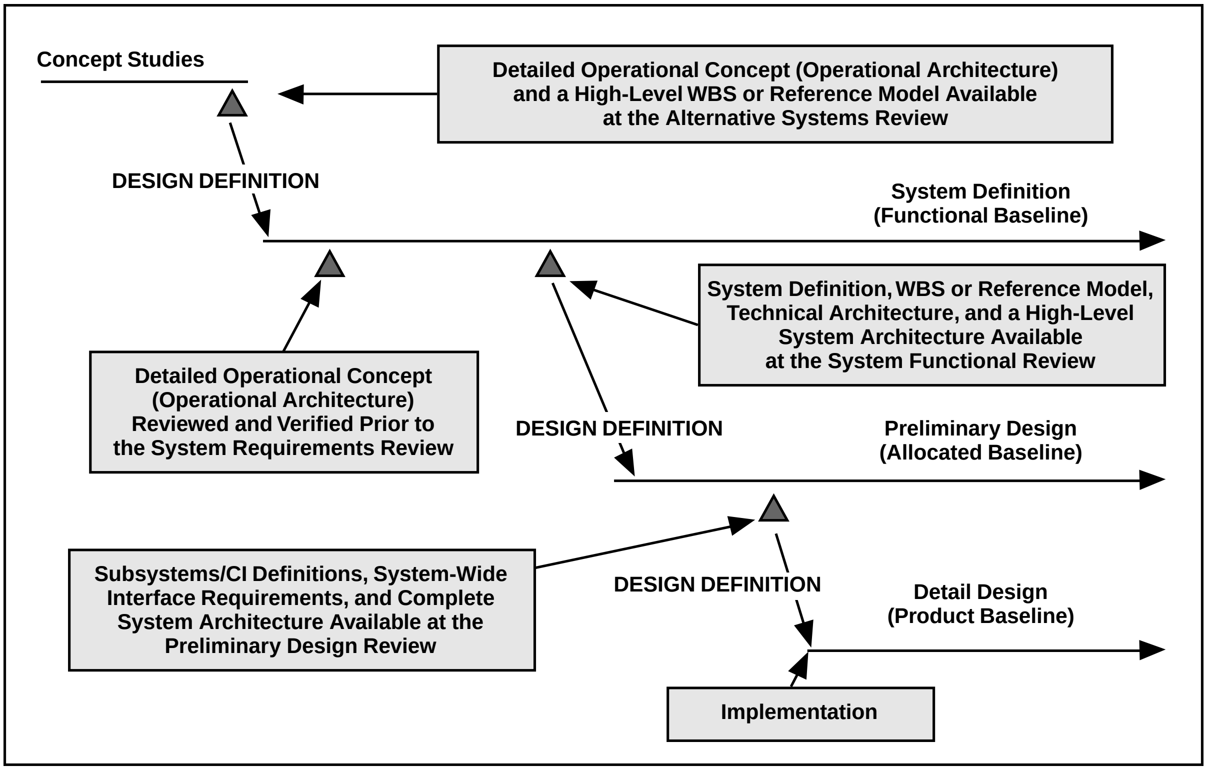


Figure 17-7. Phasing of Open System Development

end of the system definition phase (along with other functional baseline documentation). Successful completion of these items is required to perform the preliminary design, and they should be available for the System Functional Review, also referred to as the System Definition Review or System Design Review. The open system documentation can be separate or incorporated in other functional baseline documentation. The criteria for acceptance should be established in the systems engineering management plan as phase-exit criteria.

Preliminary Design Phase

Along with other allocated baseline documentation, the interface definitions should be updated and the open-system architecture completed by the end of the preliminary design effort. This documentation should also identify the proper level of openness (that is, the level of system decomposition at which the open interfaces are established) to obtain the maximum cost and logistic advantage available from industry practice.

The preliminary design establishes performance-based descriptions of the system components, as well as the interface and structure designs that integrate those components. It is in this phase that the open system approach has the most impact. Interface control should be enhanced and focused on developing modular designs that allow for maximum interchange of competitive commercial products. Review of the technical architecture (or interface definitions) becomes a key element of requirements analysis, open system focused functional partitioning becomes a key element of functional analysis and allocation, iterative analysis of modular designs becomes a key element of design synthesis, and conformance management becomes a key element of verification. Open system related products, such as the technical architecture, interface management documentation, and conformance management documentation, should be key data reviewed at the Preliminary Design Review. Again, the criteria for acceptance should be established in the systems engineering management plan as phase-exit criteria.

Detail Design Phase

The detail design phase becomes the implementation for those parts of the system that have achieved open system status. Conformance management becomes a significant activity as commercial components are chosen to meet performance and interface requirements. Conformance and interface design testing becomes a driving activity during verification to assure an open system or subsystem has been achieved and that components selected meet interface requirements and/or standards.

Systems Engineering Process

The systems engineering problem solving process consists of process steps and loops supported by system analysis and control tools. The focus of the open systems engineering process is compartmentalized design, flexible interfaces, recognized interface standards, standard components with recognized common interfaces, use of commercial and NDIs, and an increased emphasis on interface control. As shown by Figure 17-8, the open-system approach complements the systems engineering process to provide an upgradeable design.

Requirements analysis includes the review and update of interface standards and other interface definitions generated as output from previous systems engineering processes. Functional analysis and allocation focuses on functional partitioning to identify functions that can be performed independent of each other in order to minimize functional interfaces. Design synthesis focuses on modular design with open interfaces, use of open standards compliant commercial products, and the development of performance and interface specifications. The verification processes include conformance testing to validate the interface requirements are appropriate and to verify components chosen to implement the design meet the interface requirements. Engineering open designs, then, does not alter the fundamental practices within systems engineering, but, rather, provides a specific focus to the activities within that process.

System Engineering Control:
Interface Management

The key to the open systems engineering process is interface management. Interface management should be done in a more formal and comprehensive manner to rigidly identify all interfaces and

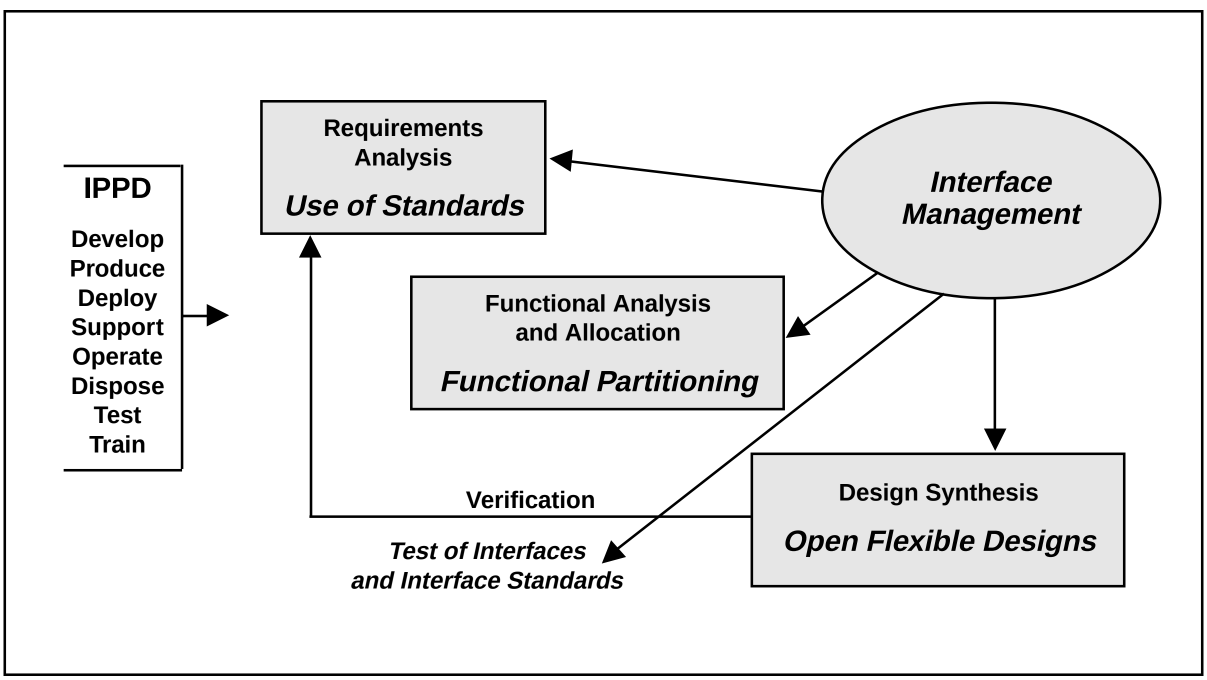


Figure 17-8. Open System Approach to the Systems Engineering Process

control the flowdown and integration of interface requirements. The interfaces become controlled elements of the baseline equal to (or considered part of) the configuration. Open system interface management emphasizes the correlation of interface requirements between interfacing systems. (Do those designing the interfacing systems understand the interface requirements in the same way?) Computer-Aided System Engineering (CASE) generated schematic block diagrams can be used to track interface design activity.

An open system is also characterized by multiple design solutions within the interfaces with emphasis on leveraging best commercial practice. The interface management effort must control interface design such that interfaces specifically chosen for an open system approach are designed based on the following priority:

- Open standards that allow competitive products,
- Open interface design that allows installation of competitive products with minimal change,
- Open interface design that allows minimal change installation of commercial or NDI products currently or planned to be in DoD use, and last,
- Unique design with interfaces designed with upgrade issues considered.

Note that these are clear priorities, not options.

Level of Openness

The level at which the interface design should focus on openness is also a consideration. Each system may have several levels of openness depending on the complexity of the system and the differences in the technology within the system. The level chosen to define the open interfaces should be supported by industry and be consistent with program objectives. For example, for most digital electronics that level is the line-replaceable (LRU) and shop-replaceable (SRU) level. On the other hand the Joint Strike Fighter intends to establish openness at a very high subsystem level to achieve a major program objective, development of different planes using common building blocks (which, in essence, serve as the reference model for the family of aircraft). The open system approach designed segments of a larger system could have additional openness at a lower level. For example, the Advanced Amphibious Assault Vehicle (AAAV) engine compartment is an open approach design allowing for different engine installation and future upgrade capability. On a lower level within the compartment the fuel filters, lines, and connectors are defined by open standard based interfaces. Other systems will define openness at other levels. Program objectives (such as interoperability, upgrade capability, cost-effective support, affordability, and risk reduction) and industry practice (based on market research) drive the choice of the level of openness that will best assure optimum utility and availability of the open system approach.

Life Cycle Considerations

Life cycle integration is established primarily through the use of integrated teaming that combines the design and life cycle planning. The major impacts on life-cycle activity include:

- ***Time and cost to upgrade a system is reduced.*** It is common in defense systems, which have average life spans in excess of 40 years, that they will require upgrade in their life due to obsolescence of original components, threat increase, and technology push that increases economy or performance. (Most commercial products are designed for a significantly shorter life than military systems, and designs that rely on these commercial products must expect that original commercial components will not necessarily be available throughout the system's life cycle.) By using an open system approach the ability to upgrade a system by changing a single or set of components is greatly enhanced. In addition, the open system approach eases the design problem of replacing the component, thereby reducing the cost and schedule of upgrade, which in turn reduces the operational impact.

- ***An open system approach enhances the use of competitive products to support the system.*** This flexibility tends to reduce the cost associated with supply support, but more importantly improves component and parts availability.

- ***Conformance management becomes a part of the life cycle configuration process.*** Replacement of components in an open system must be more controlled because the government has to control the system configuration without controlling the detail component configuration (which will come from multiple sources, all with different detail configurations). The government must expect that commercial suppliers will control the design of their components without regard to the government's systems. The government therefore must use performance- and interface-based specifications to assure the component will provide service equivalent to that approved through the acquisition process. Conformance management is the process that tracks the interface requirements through the life cycle, and assures that the new product meets those requirements.

Summary Comments

Open system design is not only compatible with systems engineering; it represents an approach that enhances the overall systems engineering effort. It controls interfaces comprehensively, provides interface visibility, reduces risk through multiple design solutions, and insists on life cycle interface control. This emphasis on interface identification and control improves systems engineers' capability to integrate the system, probably one of the hardest jobs they have. It also improves the tracking of interface requirements flow down, another key job of the systems engineer. Perhaps most importantly, this rigorous interface management improves systems engineers' ability to correctly determine where commercial items can be properly used.

CHAPTER 18

ORGANIZING AND INTEGRATING SYSTEM DEVELOPMENT

18.1 INTEGRATED DEVELOPMENT

DoD has, for years, required that system designs be integrated to balance the conflicting pressure of competing requirements such as performance, cost, supportability, producibility, and testability. The use of multi-disciplinary teams is the approach that both DoD and industry increasing have taken to achieve integrated designs. Teams have been found to facilitate meeting cost, performance, and other objectives from product concept through disposal.

The use of multi-disciplinary teams in design is known as Integrated Product and Process Development, simultaneous engineering, concurrent engineering, Integrated Product Development, Design-Build, and other proprietary and non-proprietary names expressing the same concept. (The DoD use of the term Integrated Product and Process Development (IPPD) is a wider concept that includes the systems engineering effort as an element. The DoD policy is explained later in this chapter.) Whatever name is used, the fundamental idea involves multi-functional, integrated teams (preferably co-located), that jointly derive requirements and schedules that place equal emphasis on product and process development. The integration requires:

- Inclusion of the eight primary functions in the team(s) involved in the design process,
- Technical process specialties such as quality, risk management, safety, etc., and
- Business processes (usually in an advisory capacity) such as, finance, legal, contracts, and other non-technical support.

Benefits

The expected benefits from team-based integration include:

- Reduced rework in design, manufacturing, planning, tooling, etc.,
- Improved first time quality and reduction of product variability,
- Reduced cost and cycle time,
- Reduced risk,
- Improved operation and support, and
- General improvement in customer satisfaction and product quality throughout its life cycle.

Characteristics

The key attributes that characterize a well integrated effort include:

- Customer focus,
- Concurrent development of products and processes,
- Early and continuous life cycle planning,
- Maximum flexibility for optimization,
- Robust design and improved process capability,
- Event-driven scheduling,
- Multi-disciplinary teamwork,

- Empowerment,
- Seamless management tools, and
- Proactive identification and management of risk.

Organizing for System Development

Most DoD program offices are part of a Program Executive Office (PEO) organization that is usually supported by a functional organization, such as a systems command. Contractors and other government activities provide additional necessary support. Establishing a system development organization requires a network of teams that draw from all these organizations. This network, sometimes referred to as the enterprise, represents the interests of all the stakeholders and provides vertical and horizontal communications.

These integrated teams are structured using the WBS and designed to provide the maximum vertical and horizontal communication during the development process. Figure 18-1 shows how team structuring is usually done. At the system level there is usually a management team and a design team. The management team would normally consist of the government and contractor program managers, the deputy program manager(s), possibly the contractor Chief Executive Officer, the contracting officer, major advisors picked by the program manager, the system design team leader, and other key members of the system design team. The design team usually consists of the first-level subsystem and life-cycle integrated team leaders.

The next level of teams is illustrated on Figure 18-1 as either product or process teams. These teams are responsible for designing system segments (product teams) or designing the supporting or enabling products (process teams). At this level the process teams are coordinating the system level process development. For example, the support team will integrate the supportability analysis from the parts being generated in lower-level design and

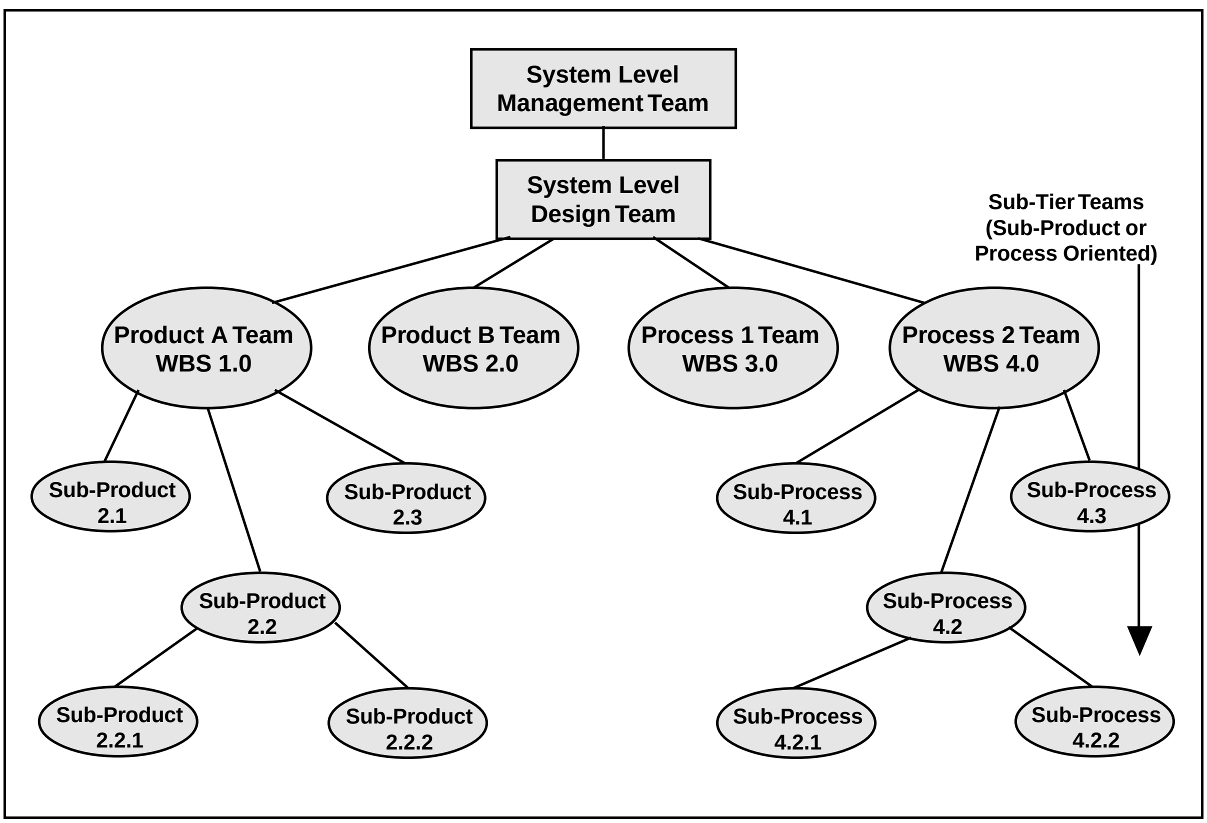


Figure 18-1. Integrated Team Structure

support process teams. Teams below this level continue the process at a lower level of decomposition. Teams are formed only to the lowest level necessary to control the integration. DoD team structures rarely extend lower than levels three or four on the WBS, while contractor teams may extend to lower levels, depending on the complexities of the project and the approach favored by management.

The team structure shown by Figure 18-1 is a hierarchy that allows continuous vertical communication. This is achieved primarily by having the team leaders, and, if appropriate, other key members of a team, be team members of the next highest team. In this manner the decisions of the higher team is immediately distributed and explained to the next team level, and the decisions of the lower teams are presented to the higher team on a regular basis. Through this method decisions of lower-level teams follow the decision making of higher teams, and the higher-level teams' decisions incorporate the concerns of lower-level teams.

The normal method to obtain horizontal communication is shown in Figure 18-2. At least one team member from the Product A Team is also a member of the Integration and Test Team. This member would have a good general knowledge of both testing and Product A. The member's job would be to assist the two teams in designing their end or enabling products, and in making each understand how their decisions would impact the other team. Similarly, the member that sits on both Product A and B teams would have to understand the both technology and the interface issues associated with both items.

The above is an idealized case. Each type of system, each type of contractor organization, and each level of available resources requires a tailoring of this structure. With each phase the focus and the tasks change and so should the structure. As phases

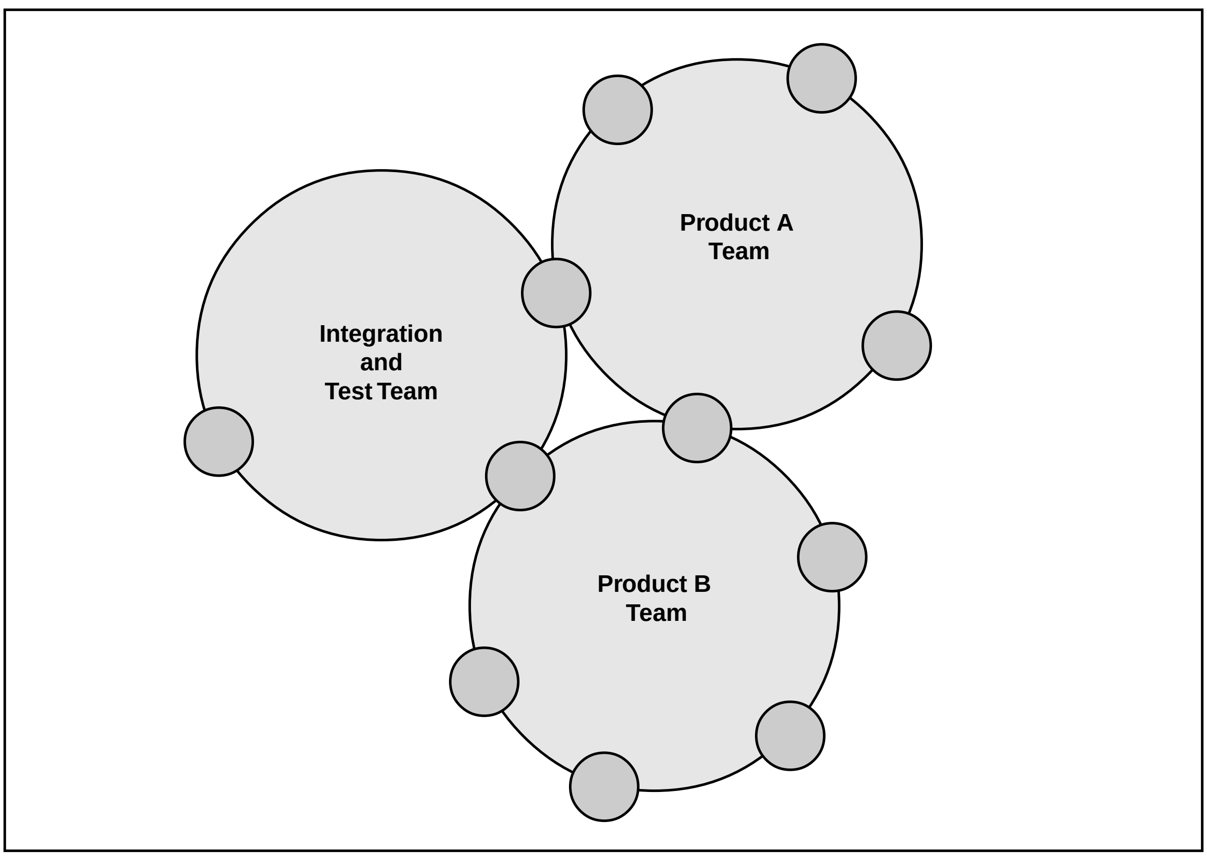


Figure 18-2. Cross Membership

are transited, the enterprise structure and team membership should be re-evaluated and updated.

18.2 INTEGRATED TEAMS

Integrated teams are composed of representatives from all appropriate primary functional disciplines working together with a team leader to:

- Design successful and balanced products,
- Develop the configuration for successful life-cycle control,
- Identify and resolve issues, and
- Make sound and timely decisions.

The teams follow the disciplined approach of the systems engineering process starting with requirements analysis through to the development of configuration baselines as explained earlier in this book. The system-level design team should be responsible for systems engineering management planning and execution. The system-level management team, the highest level program IPT, is responsible for acquisition planning, resource allocation, and management. Lower-level teams are responsible for planning and executing their own processes.

Team Organization

Good teams do not just happen; they are the result of calculated management decisions and actions. Concurrent with development of the enterprise organization discussed above, each team must also be developed. Basically the following are key considerations in planning for a team within an enterprise network:

- The team must have appropriate representation from the primary functions, technical specialties, and business support,
- There must be links to establish vertical and horizontal communication in the enterprise,
- You should limit over-uses of cross membership. Limit membership on three or four teams as a rough rule of thumb for the working level, and
- Ensure appropriate representation of government, contractor, and vendors to assure integration across key organizations.

Team Development

When teams are formed they go through a series of phases before a synergistic self-actuating team is evolved. These phases are commonly referred to as forming, storming, norming and performing. The timing and intensity of each phase will depend on the team size, membership personality, effectiveness of the team building methods employed, and team leadership. The team leaders and an enterprise-level facilitator provide leadership during the team development.

Forming is the phase where the members are introduced to their responsibilities and other members. During this period members will tend to need a structured situation with clarity of purpose and process. If members are directed during this initial phase, their uncertainty and therefore apprehension is reduced. Facilitators controlling the team building should give the members rules and tasks, but gradually reduce the level of direction as the team members begin to relate to each other. As members become more familiar with other members, the rules, and tasks, they become more comfortable in their environment and begin to interact at a higher level.

This starts the storming phase. *Storming* is the conflict brought about by interaction relating to the individuals' manner of dealing with the team tasks and personalities. Its outcome is members who understand the way they have to act with other members to accomplish team objectives. The dynamics of storming can be very complex and intense, making it the critical phase. Some teams will go through it quickly without a visible ripple, others will be loud and hot, and some will never emerge from this phase. The team building facilitators must be alert to dysfunctional activity.

Members may need to be removed or teams reorganized. Facilitators during this period must act as coaches, directing but in a personal collaborative way. They should also be alert for members that are avoiding storming, because the team will not mature if there are members who are not personally committed to participate in it.

Once the team has learned to interact effectively it begins to shape its own processes and become more effective in joint tasks. It is not unusual to see some reoccurrence of storming, but if the storming phase was properly transitioned these incidences should be minor and easily passed. In this phase, *norming*, the team building facilitators become a facilitator to the team—not directing, but asking penetrating questions to focus the members. They also monitor the teams and correct emerging problems.

As the team continues to work together on their focused tasks, their performance improves until they reach a level of self-actuation and quality decision making. This phase, *performing*, can take a while to reach, 18 months to two years for a system-level design team would not be uncommon. During the performing stage, the team building facilitator monitors the teams and corrects emerging problems.

At the start of a project or program effort, team building is commonly done on an enterprise basis with all teams brought together in a team-building exercise. There are two general approaches to the exercise:

- A team-learning process where individuals are given short but focused tasks that emphasize group decision, trust, and the advantages of diversity.
- A group work-related task that is important but achievable, such as a group determination of the enterprise processes, including identifying and removing non-value added traditional processes.

Usually these exercises allow the enterprise to pass through most of the storming phase if done correctly. Three weeks to a month is reasonable for this process, if the members are in the same location. Proximity does matter and the team building and later team performance are typically better if the teams are co-located.

18.3 TEAM MAINTENANCE

Teams can be extremely effective, but they can be fragile. The maintenance of the team structure is related to empowerment, team membership issues, and leadership.

Empowerment

The term empowerment relates to how responsibilities and authority is distributed throughout the enterprise. Maintenance of empowerment is important to promote member ownership of the development process. If members do not have personal ownership of the process, the effectiveness of the team approach is reduced or even neutralized. The quickest way to destroy participant ownership is to direct, or even worse, overturn solutions that are properly the responsibility of the team. The team begins to see that the responsibility for decisions is at a higher level rather than at their level, and their responsibility is to follow orders, not solve problems.

Empowerment requires:

- The flow of authority through the hierarchy of teams, not through personal direction (irrespective of organizational position). Teams should have clear tasking and boundaries established by the higher-level teams.
- Responsibility for decision making to be appropriate for the level of team activity. This requires management and higher-level teams to be specific, clear, complete, and comprehensive in establishing focus and tasking, and in specifying what decisions must be coordinated with higher levels. They should then avoid imposing or overturning decisions more properly in the realm of a lower level.

- Teams at each level be given a clear understanding of their duties and constraints. Within the bounds of those constraints and assigned duties members should have autonomy. Higher-level teams and management either accept their decisions, or renegotiate the understanding of the task.

Membership Issues

Another maintenance item of import is team member turnover. Rotation of members is a fact of life, and a necessary process to avoid teams becoming too closed. However, if the team has too fast a turnover, or new members are not fully assimilated, the team performance level will decline and possibly revert to storming. The induction process should be a team responsibility that includes the immediate use of the new team member in a jointly performed, short term, easily achievable, but important task.

Teams are responsible for their own performance, and therefore should have significant, say over the choice of new members. In addition teams should have the power to remove a member; however, this should be preceded by identification of the problem and active intervention by the facilitator. Removal should be a last resort.

Awards for performance should, where possible, be given to the team rather than individuals (or equally to all individuals on the team). This achieves several things: it establishes a team focus, shows recognition of the team as a cohesive force, recognizes that the quality of individual effort is at least in part due to team influence, reinforces the membership's dedication to team objectives, and avoids team member segregation due to uneven awards. Some variation on this theme is appropriate where different members belong to different organizations, and a common award system does not exist. The system-level management team should address this issue, and where possible assure equitable awards are given team members. A very real constraint on cash awards in DoD rises in the case of teams that include both civilian and military members. Military members cannot be given cash awards, while civilians can. Con-sequently, managers must actively seek ways to reward all team members appropriately, leaving no group out at the expense of others.

Leadership

Leadership is provided primarily by the organizational authority responsible for the program, the enterprise facilitator, and the team leaders. In a DoD program, the organizational leaders are usually the program manager and contractor senior manager. These leaders set the tone of the enterprise adherence to empowerment, the focus of the technical effort, and the team leadership of the system management team. These leaders are responsible to see that the team environment is maintained. They should coordinate their action closely with the facilitator.

Facilitators

Enterprises that have at least one facilitator find that team and enterprise performance is easier to maintain. The facilitator guides the enterprise through the team building process, monitors the team network through metrics and other feedback, and makes necessary corrections through facilitation. The facilitator position can be:

- A separate position in the contractor organization,
- Part of the responsibilities of the government systems engineer or contractor project manager, or
- Any responsible position in the first level below the above that is related to risk management.

Obviously the most effective position would be one that allows the facilitator to concentrate on the teams' performance. Enterprise level facilitators should have advanced facilitator training and (recommended) at least a year of mentored experience. Facilitators should also have significant broad experience in the technical area related to the development.

Team Leaders

The team leaders are essential for providing and guiding the team focus, providing vertical communication to the next level, and monitoring the team's performance. Team leaders must have a clear picture of what constitutes good performance for their team. They are not supervisors, though in some organizations they may have supervisory administrative duties. The leader's primary purpose is to assure that the environment is present that allows the team to perform at its optimum level—not to direct or supervise.

The team leader's role includes several difficult responsibilities:

- Taking on the role of coach as the team forms,
- Facilitating as the team becomes self-sustaining,
- Sometimes serving as director (only when a team has failed, needs refocus or correction, and is done with the facilitator),
- Providing education and training for members,
- Facilitating team learning,
- Representing the team to upper management and the next higher-level team, and
- Facilitating team disputes.

Team leaders should be trained in basic facilitator principles. This training can be done in about a week, and there are numerous training facilities or companies that can offer it.

18.4 TEAM PROCESSES

Teams develop their processes from the principles of system engineering management as presented earlier in the book. The output of the teams is the design documentation associated with products identified on the system architecture, including both end product components and enabling products.

Teams use several tools to enhance their productivity and improve communication among enterprise members. Some examples are:

- Constructive modeling (CAD/CAE/CAM/CASE) to enhance design understanding and control,
- Trade-off studies and prioritization,
- Event-driven schedules,
- Prototyping,
- Metrics, and most of all
- Integrated membership that represents the life cycle stakeholders.

Integrated Team Rules

The following is a set of general rules that should guide the activities and priorities of teams in a system design environment:

- Design results must be communicated clearly, effectively, and timely.
- Design results must be compatible with initially defined requirements.
- Continuous "up-the-line" communication must be institutionalized.
- Each member needs to be familiar with all system requirements.
- Everyone involved in the team must work from the same database.
- Only one member of the team has the authority to make changes to one set of master documentation.
- All members have the same level of authority (one person, one vote).
- Team participation is consistent, success-oriented, and proactive.

- Team discussions are open with no secrets.
- Team member disagreements must be reasoned disagreement (alternative plan of action versus unyielding opposition).
- Trade studies and other analysis techniques are used to resolve issues.
- Issues are raised and resolved early.
- Complaints about the team are not voiced outside the team. Conflicts must be resolved internally.

Guidelines for Meeting Management

Even if a team is co-located as a work unit, regular meetings will be necessary. These meetings and their proper running become even more important if the team is not co-located and the meeting is the primary means of one-on-one contact. A well-run technical meeting should incorporate the following considerations:

- Meetings should be held only for a specific purpose and a projected duration should be targeted.
- Advance notice of meetings should normally be at least two weeks to allow preparation and communication between members.
- Agendas, including time allocations for topics and supportive material should be distributed no less than three business days before the team meeting. The objective of the meeting should be clearly defined.
- Stick to the agenda during the meeting. Then cover new business. Then review action items.
- Meeting summaries should record attendance, document any decision or agreements reached, document action items and associated due-dates, provide a draft agenda for the next meeting, and frame issues for higher-level resolution.
- Draft meeting summaries should be provided to members within one working day of the meeting. A final summary should be issued within two working days after the draft comments deadline.

18.5 BARRIERS TO INTEGRATION

There are numerous barriers to building and maintaining a well functioning team organization, and they are difficult to overcome. Any one of these barriers can negate the effectiveness of an integrated development approach. Common barriers include:

- Lack of top management support,
- Team members not empowered,
- Lack of access to a common database,
- Lack of commitment to a cultural change,
- Functional organization not fully integrated into a team process,
- Lack of planning for team effort,
- Staffing requirements conflict with teams,
- Team members not collocated,
- Insufficient team education and training,
- Lessons learned and successful practices not shared across teams,
- Inequality of team members,
- Lack of commitment based on perceived uncertainty,
- Inadequate resources, and
- Lack of required expertise on either the part of the contractor or government.

Breaking Barriers

Common methods to combat barriers include:

- Education and training, and then more education and training: it breaks down the uncertainty of change, and provides a vision and method for success.
- Use a facilitator not only to build and maintain teams, but also to observe and advise management.
- Obtain management support up front. Management must show leadership by managing the teams' environment rather than trying to manage people.
- Use a common database open to all enterprise members.
- Establish a network of teams that integrates the design and provides horizontal and vertical communication.
- Establish a network that does not over-tax available resources. Where a competence is not available in the associated organizations, hire it through a support contractor.
- Where co-location is not possible have regular working sessions of several days duration. Telecommunications, video conferencing, and other technology based techniques can also go far to alleviate the problems of non-collocation.

Summary Comments

- Integrating system development is a systems engineering approach that integrates all essential primary function activities through the use of multi-disciplinary teams, to optimize the design, manufacturing and supportability processes.
- Team building goes through four phases: forming, storming, norming, and performing.
- Key leadership positions in a program network of teams are the program manager, facilitator, and team leaders.
- A team organization is difficult to build and maintain. It requires management attention and commitment over the duration of the teams involved.

SUPPLEMENT 18-A

IPPD – A DOD MANAGEMENT PROCESS

The DoD policy of Integrated Product and Process Development (IPPD) is a broad view of integrated system development which includes not only systems engineering, but other areas involved in formal decision making related to system development. DoD policy emphasizes integrated management at and above the Program Manager (PM) level. It requires IPPD at the systems engineering level, but does not direct specific organizational structures or procedures in recognition of the need to design a tailored IPPD process to every individual situation.

Integrated Product Teams

One of the key IPPD tenets is multi-disciplinary integration and teamwork achieved through the use of Integrated Product Teams (IPTs). While IPTs may not be the best solution for every management situation, the requirement to produce integrated designs that give consideration to a wide array of technical and business concerns leads most organizations to conclude that IPTs are the best organizational approach to systems management. PMs should remember that the participation of a contractor or a prospective contractor on a IPT should be in accordance with statutory requirements, such as procurement integrity rules. The service component's legal advisor must review prospective contractor involvement on IPTs. To illustrate issues the government-contractor team arrangement raises, the text box at the end of this section lists nine rules developed for government members of the Advanced Amphibious Assault Vehicle (AAAV) design IPTs.

The Secretary of Defense has directed that DoD perform oversight and review by using IPTs. These IPTs function in a spirit of teamwork with participants empowered and authorized, to the maximum extent possible, to make commitments for the organization or the functional area they represent. IPTs are composed of representatives from all appropriate functional disciplines working together to build successful programs and enabling decision makers to make the right decisions at the right time.

DoD IPT Structure

The DoD oversight function is accomplished through a hierarchy of teams that include levels of management from DoD to the program level. There are three basic levels of IPTs: the Overaching IPT (OIPT), the Working IPTs (WIPT), and Program IPTs with the focus and responsibilities as shown by Figure 18-3. For each ACAT I program, there will be an OIPT and at least one WIPT. WIPTs will be developed for particular functional topics, e.g., test, cost/performance, contracting, etc. An Integrating IPT (IIPT) will coordinate WIPT efforts and cover all topics not otherwise assigned to another IPT. These teams are structurally organized as shown on Figure 18-4.

Overarching IPT (OIPT)

The OIPT is a DoD level team whose primary responsibility is to advise the Defense Acquisition Executive on issues related to programs managed at that level. The OIPT membership is made up of the principals that are charged with responsibility for the many functional offices at the Office of the Secretary of Defense (OSD).

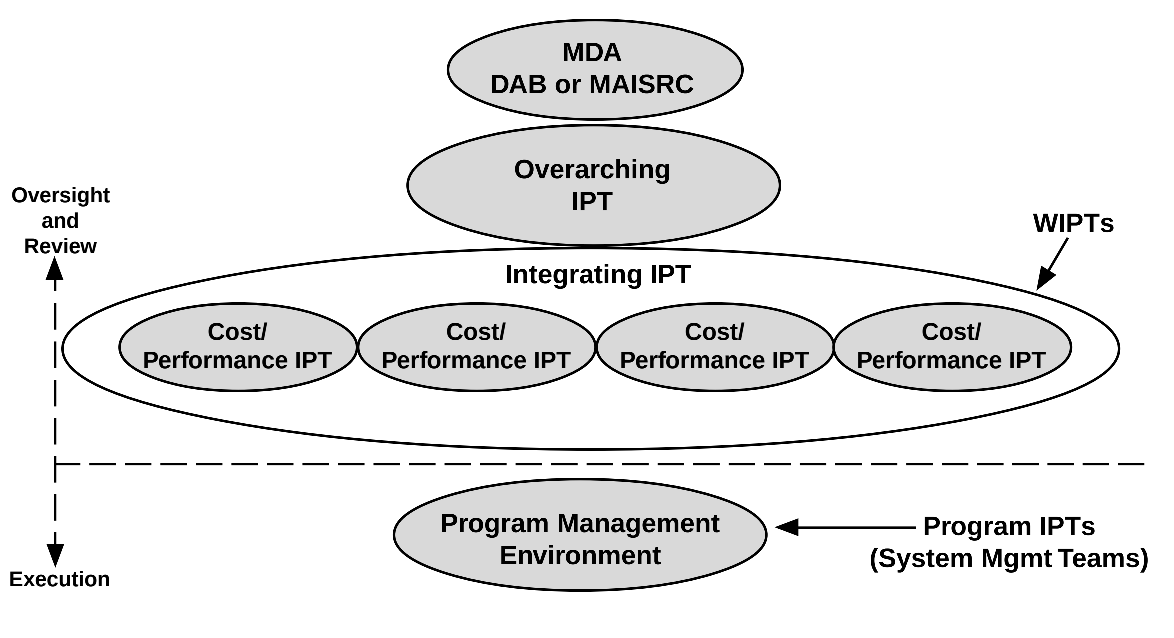
The OIPT provides:

- Top-level strategic guidance,

Organization	Teams	Focus	Participant Responsibilities
OSD and Components	OIPT*	• Strategic Guidance • Tailoring • Program Assessment • Resolve Issues Elevated by WIPTs	• Program Success • Functional Area Leadership • Independent Assessment • Issue Resolution
	WIPTs*	• Planning for Program Success • Opportunities for Acquisition Reform (e.g. innovation, streamlining) • Identify/Resolve Program Issues • Program Status	• Functional Knowledge and Experience • Empowered Contribution • Recom.'s for Program Success • Communicate Status and Unresolved Issues
Program Teams and System Contractors	Program IPTs**	• Program Execution • Identify and Implement Acquisition Reform	• Manage Complete Scope of Program Resources, and Risk • Integrate Government and Contractor Efforts for Report Program Status and Issues

* Covered in "Rules of the Road"
** Covered in "Guide to Implementation and Management of IPPD in DoD Acquisition"

Figure 18-3. Focus and Responsibilities of IPTs



Extracted from *Rules of the Road, A Guide for Leading Successful Integrated Product Teams.*

Figure 18-4. IPT Structure

- Functional area leadership,
- Forum for issue resolution,
- Independent assessment to the MDA,
- Determine decision information for next milestone review, and
- Provide approval of the WIPT structures and resources.

Working-Level IPT (WIPT)

The WIPTs may be thought of as teams that link the PM to the OIPT. WIPTs are typically functionally specialized teams (test, cost-performance, etc.). The PM is the designated head of the WIPT, and membership typically includes representation from various levels from the program to OSD staff. The principal functions of the WIPT are to advise the PM is the area of specialization and to advise the OIPT of program status.

The duties of the WIPT include:

- Assisting the PM in developing strategies and in program planning, as requested by the PM,
- Establishing IPT plan of action and milestones,
- Proposing tailored document and milestone requirements,
- Reviewing and providing early input to documents,
- Coordinating WIPT activities with the OIPT members,
- Resolving or evaluating issues in a timely manner, and
- Obtaining principals' concurrence with applicable documents or portions of documents.

Program IPTs

Program IPTs are teams that perform the program tasks. The integration of contractors with the government on issues relative to a given program truly occurs at the program IPT level. The development teams (product and process teams) described earlier in this chapter would be considered program IPTs. Program IPTs would also include teams formed for business reasons, for example teams established to prepare Planning, Programming, and Budgeting System (PPBS) documentation, to prepare for Milestone Approval, to develop the RFP, or the like.

SUPPLEMENT 18-B

GOVERNMENT ROLE ON IPTs

The following list was developed by the Advanced Amphibious Assault Vehicle (AAAV) program to inform its government personnel of their role on contractor/government integrated teams. It addresses government responsibilities and the realities imposed by contractual and legal constraints. Though it is specific to the AAAV case, it can be used as guidance in the development of team planning for other programs.

1. The IPTs are contractor-run entities. We do not lead or manage the IPTs.

2. We serve as "customer" representatives on the IPTs. We are there to REDUCE THE CYCLE TIME of contractor-Government (customer) communication. In other words, we facilitate contractor personnel getting Government input faster. Government IPT members also enable us to provide the contractor IPT Status and issue information up the Government chain on a daily basis (instead of monthly or quarterly).

3. WE DO NOT DO the contractor's IPT WORK, or any portion of their work or tasks. The contractor has been contracted to perform the tasks outlined in the contract SOW; their personnel and their subcontractors' personnel will perform those tasks, not us. But Government IPT members will be an active part of the deliberations during the development of, and participate in "on-the-fly" reviews of deliverables called out in CDRLs.

4. When asked by contractor personnel for the Government's position or interpretation, Government IPT members can offer their personal opinion, as an IPT member, or offer expert opinion; you can provide guidance as to our "customer" opinion and what might be acceptable to the Government but you can only offer the "Government" position for items that have been agreed to by you and your Supervisor. IT IS UP TO YOUR SUPERVISORS TO EMPOWER EACH OF YOU TO AN APPROPRIATE LEVEL OF AUTHORITY. It is expected that this will start at a minimal level of authority and be expanded as each individual's IPT experience and program knowledge grows. However… (see items 5 and 6).

5. Government IPT members CAN NOT authorize any changes or deviations to/from the contract SOW or Specifications. Government IPT members can participate in the deliberations and discussions that would result in the suggestion of such changes. If/When an IPT concludes that the best course of action is not in accordance with the contract, and a contract change is in order, then the contractor must submit a Contract Change Request (CCR) through normal channels.

6. Government IPT members CAN NOT authorize the contractor to perform work that is in addition to the SOW/contract requirements. The contractor IPTs can perform work that is not specifically required by the contract, at their discretion (provided they stay within the resources as identified in the Team Operating Contract (TOC).

7. Government IPT member participation in contractor IPT activities IS NOT Government consent that the work is approved by the Government or is chargeable to the contract. If an IPT is doing something questionable, identify it to your supervisor or Program Management Team (PMT) member.

8. Government members of IPTs do not approve or disapprove of IPT decisions, plans, or reports. You offer your opinion in their development, you vote as a member, and you coordinate issues with your Supervisor and bring the "Government" opinion (in the form of your opinion) back to the IPT, with the goal of improving the quality of the products; you don't have veto power.

9. Government IPT members are still subject to all the Government laws and regulations regarding "directed changes," ethics, and conduct. Your primary function is to perform those functions that are best done by Government employees, such as:

- Conveying to contractor personnel your knowledge/expertise on Marine Corps operations and maintenance techniques;

- Interfacing with all other Government organizations (e.g., T&E);

- Control/facilitization of government furnished equipment and materials (GFE and GFM);

- Ensuring timely payment of submitted vouchers; and

- Full participation in Risk Management.

CHAPTER 19

CONTRACTUAL CONSIDERATIONS

19.1 INTRODUCTION

This chapter describes how the systems engineer supports the development and maintenance of the agreement between the project office and the contractor that will perform or manage the detail work to achieve the program objectives. This agreement has to satisfy several stakeholders and requires coordination between responsible technical, managerial, financial, contractual, and legal personnel. It requires a document that conforms to the Federal Acquisition Regulations (and supplements), program PPBS documentation, and the System Architecture. As shown by Figure 19-1, it also has to result in a viable cooperative environment that allows necessary integrated teaming to take place.

The role of technical managers or systems engineers is crucial to satisfying these diverse concerns. Their primary responsibilities include:

- Supporting or initiating the planning effort. The technical risk drives the schedule and cost risks which in turn should drive the type of contractual approach chosen,
- Prepares or supports the preparation of the source selection plan and solicitation clauses concerning proposal requirements and selection criteria,
- Prepares task statements,

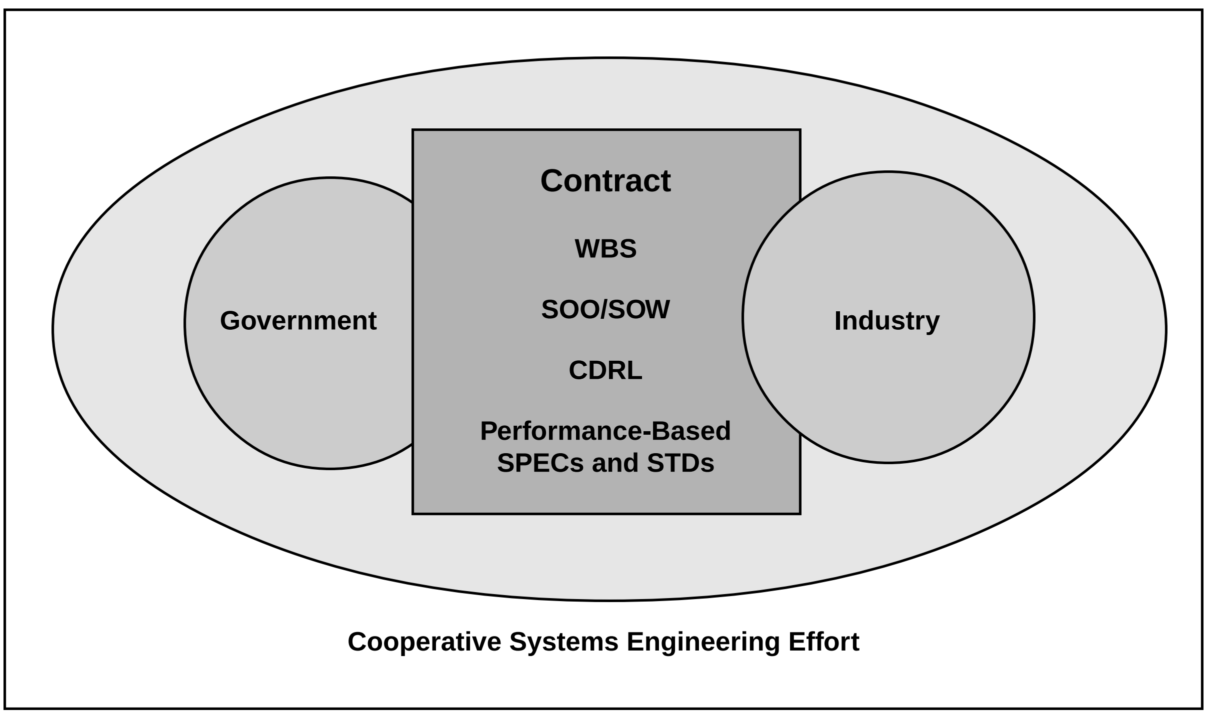


Figure 19-1. Contracting Process

- Prepares the Contract Data Requirements List (CDRL),
- Supports negotiation and participates in source selection evaluations,
- Forms Integrated Teams and coordinates the government side of combined government and industry integrated teams,
- Monitors the contractor's progress, and
- Coordinates government action in support of the contracting officer.

This chapter reflects the DoD approach to contracting for system development. It assumes that there is a government program or project office that is tasking a prime contractor in a competitive environment. However, in DoD there is variation to this theme. Some project activities are tasked directly to a government agency or facility, or are contracted sole source. The processes described in this chapter should be tailored as appropriate for these situations.

19.2 SOLICITATION DEVELOPMENT

As shown by Figure 19-2, the DoD contracting process begins with planning efforts. Planning includes development of a Request for Proposal (RFP), specifications, a Statement of Objective (SOO) or Statement of Work (SOW), a source selection plan, and the Contract Data Requirements List (CDRL).

Request for Proposal (RFP)

The RFP is the solicitation for proposals. The government distributes it to potential contractors. It describes the government's need and what the offeror must do to be considered for the contract. It establishes the basis for the contract to follow.

The key systems engineering documents included in a solicitation are:

- A statement of the work to be performed. In DoD this is a SOW. A SOO can be used to obtain a SOW or equivalent during the selection process.

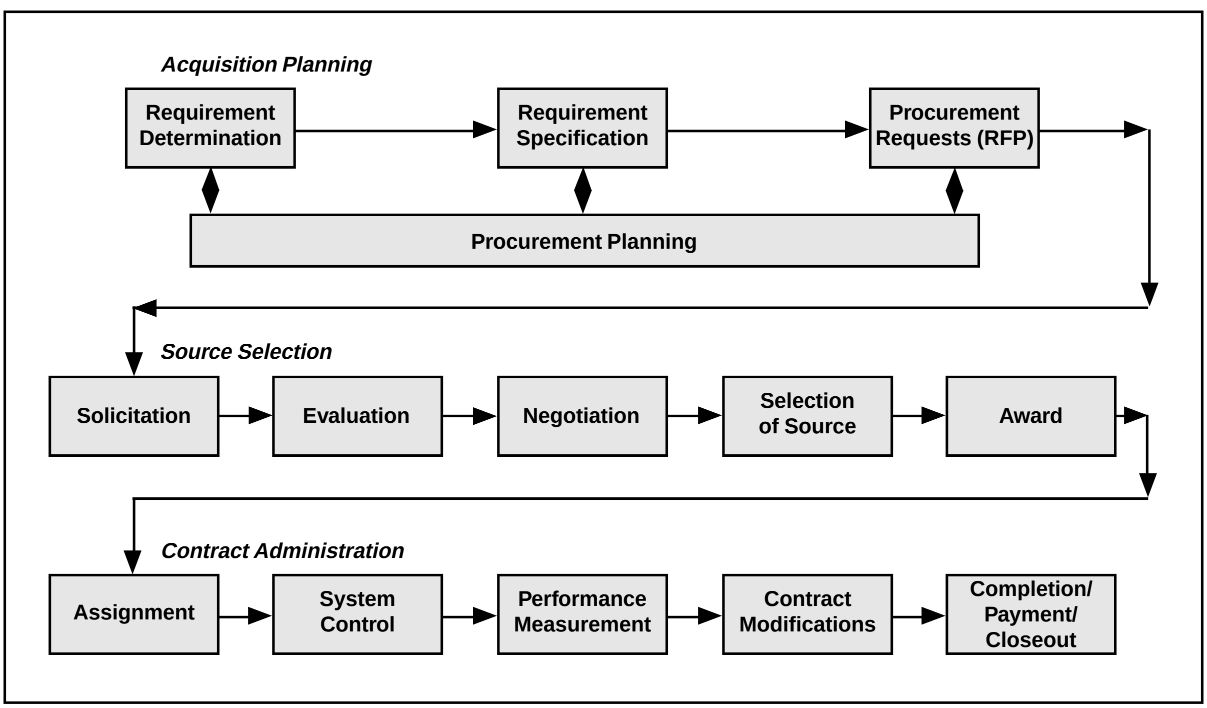


Figure 19-2. Contracting Process

- A definition of the system. Appropriate specifications and any additional baseline information necessary for clarification form this documentation. This is generated by the systems engineering process as explained earlier in this book.

- A definition of all data required by the customer. In DoD this accomplished through use of the Contract Data Requirements List (CDRL).

The information required to be in the proposals responding to the solicitation is also key for the systems engineer. An engineering team will decide the technical and technical management merits of the proposals. If the directions to the offerors are not clearly and correctly stated, the proposal will not contain the information needed to evaluate the offerors. In DoD Sections L and M of the RFP are those pivotal documents.

Task Statement

The task statement prepared for the solicitation will govern what is actually received by the government, and establish criteria for judging contractor performance. Task requirements are expressed in the SOW. During the solicitation phase the tasks can be defined in very general way by a SOO. Specific details concerning SOOs and SOWs are attached at the end of this chapter.

As shown by Figure 19-3, solicitation tasking approaches can be categorized into four basic options: use of a basic operational need, a SOO, a SOW, or a detail specification.

Option 1 maximizes contractor flexibility by submitting the Operational Requirements Document (ORD) to offerors as a requirements document (e.g. in place of SOO/SOW), and the offerors are requested to propose a method of developing a solution to the ORD. The government identifies its areas of concern in Section M (evaluation factors) of the RFP to provide guidance. Section L (instructions to the offerors) should require the bidders write a SOW based on the ORD as *part* of their proposal. The offeror proposes the type of system. The contractor develops the system specification and the Work Breakdown Structure (WBS). In general this option is appropriate for early efforts where contractor input is necessary to expand the understanding of physical solutions and alternative system approaches.

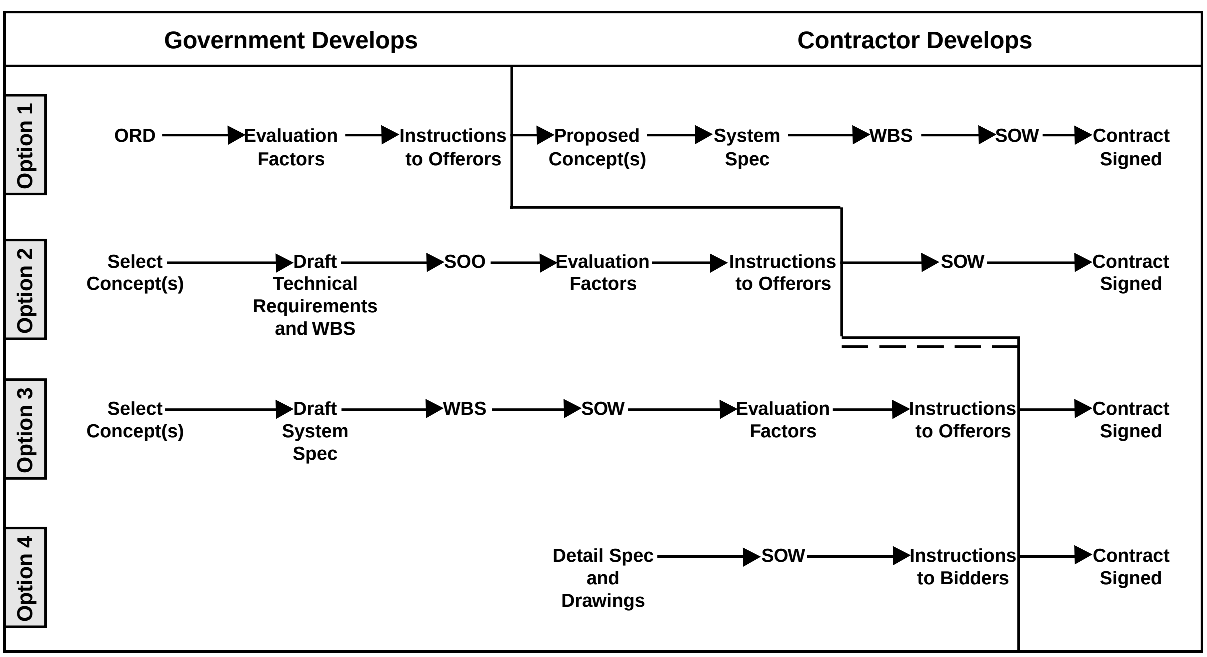


Figure 19-3. Optional Approaches

Option 2 provides moderate contractor flexibility by submitting a SOO to the offerors as the Section C task document (e.g., in place of SOW.) The government identifies its areas of concern in Section M (evaluation factors) to provide guidance. Section L (instructions to the offerors) should require as part of the proposal that offerors write a SOW based on the SOO. In this case the government usually selects the type of system, writes a draft technical-requirements document or system specification, and writes a draft WBS. This option is most appropriate when previous efforts have not defined the system tightly. The effort should not have any significant design input from the previous phase. This method allows for innovative thinking by the bidders in the proposal stage. It is a preferred method for design contracts.

Option 3 lowers contractor flexibility, and increases clarity of contract requirements. In this option the SOW is provided to the Contractor as the contractual task requirements document. The government provides instructions in Section L to the offerors to describe the information needed by the government to evaluate the contractor's ability to accomplish the SOW tasks. The government identifies evaluation factors in Section M to provide guidance for priority of the solicitation requirements. In most cases, the government selects the type of system, and provides the draft system spec, as well as the draft WBS. This option is most appropriate when previous efforts have defined the system to the lower WBS levels or where the product baseline defines the system. Specifically when there is substantial input from the previous design phase and there is a potential for a different contractor on the new task, the SOW method is appropriate.

Option 4 minimizes contractor flexibility, and requires maximum clarity and specificity of contract requirements. This option uses an Invitation for Bid (IFB) rather than an RFP. It provides bidders with specific detailed specifications or task statements describing the contract deliverables. They tell the contractor exactly what is required and how to do it. Because there is no flexibility in the contractual task, the contract is awarded based on the low bid. This option is appropriate when the government has detailed specifications or other product baseline documentation that defines the deliverable item sufficient for production. It is generally used for simple build-to-print reprocurement.

Data Requirements

As part of the development of an IFB or RFP, the program office typically issues a letter that describes the planned procurement and asks integrated team leaders and affected functional managers to identify and justify their data requirements for that contract. The data should be directly associated with a process or task the contractor is required to perform.

The affected teams or functional offices then develop a description of each data item needed. Data Item Descriptions (DIDs), located in the Acquisition Management Systems and Data Requirements Control List (AMSDL), can be used for guidance in developing these descriptions. Descriptions should be performance based, and format should be left to the contractor as long as all pertinent data is included. The descriptions are then assembled and submitted for inclusion in the solicitation. The listing of data requirements in the contract follows an explicit format and is referred to as the CDRL.

In some cases the government will relegate the data call to the contractor. In this case it is important that the data call be managed by a government/contractor team, and any disagreements be resolved prior to formal contract change incorporating data requirements. When a SOO approach is used, the contractor should be required by section L to propose data requirements that correspond to their proposed SOW.

There is current emphasis on electronic submission of contractually required data. Electronic Data Interchange (EDI) sets the standards for compatible data communication formats.

Additional information on data management, types of data, contractual considerations, and sources of data are presented in Chapters 10 and

13. Additional information on CDRLs is provided at the end of this chapter.

Technical Data Package Controversy

Maintenance of a detailed baseline such as the "as built" description of the system, usually referred to as a Technical Data Package (TDP), can be very expensive and labor intensive. Because of this, some acquisition programs may not elect to purchase this product description. If the Government will not own the TDP the following questions must be resolved prior to solicitation issue:

- What are the pros and cons associated with the TDP owned by the contractor?
- What are the support and reprocurement impacts?
- What are the product improvement impacts?
- What are the open system impacts?

In general the government should have sufficient data rights to address life cycle concerns, such as maintenance and product upgrade. The extent to which government control of configurations and data is necessary will depend on support and reprocurement strategies. This, in turn, demands that those strategic decisions be made as early as possible in the system development to avoid purchasing data rights as a hedge against the possibility that the data will be required later in the program life cycle.

Source Selection

Source Selection determines which offeror will be the contractor, so this choice can have profound impact on program risk. The systems engineer must approach the source selection with great care because, unlike many planning decisions made early in product life cycles, the decisions made relative to source selection can generally not be easily changed once the process begins. Laws and regulations governing the fairness of the process require that changes be made very carefully—and often at the expense of considerable time and effort on the part of program office and contractor personnel. In this environment, even minor mistakes can cause distortion of proper selection.

The process starts with the development of a Source Selection Plan (SSP), that relates the organizational and management structure, the evaluation factors, and the method of analyzing the offerors' responses. The evaluation factors and their priority are transformed into information provided to the offerors in sections L and M of the RFP. The offerors' proposals are then evaluated with the procedures delineated in the SSP. These evaluations establish which offerors are conforming, guide negotiations, and are the major factor in contractor selection. The SSP is further described at the end of this chapter.

The system engineering area of responsibility includes support of SSP development by:

- Preparing the technical and technical management parts of evaluation factors,
- Organizing technical evaluation team(s), and
- Developing methods to evaluate offerors' proposals (technical and technical management).

19.3 SUMMARY COMMENTS

- Solicitation process planning includes development of a Request for Proposal, specifications, a Statement of Objective or Statement of Work, a source selection plan, and the Contract Data Requirements List.
- There are various options available to program offices as far as the guidance and constraints imposed on contractor flexibility. The government, in general, prefers that solicitations be performance-based.
- Data the contractor is required to provide the government is listed on the CDRL List.
- Source Selection is based on the evaluation criteria outlined in the SSP and reflected in Sections L and M of the RFP.

SUPPLEMENT 19-A

STATEMENT OF OBJECTIVES (SOO)

The SOO is an alternative to a government prepared SOW. A SOO provides the Government's overall objectives and the offeror's required support to achieve the contractual objectives. Offerors use the SOO as a basis for preparing a SOW which is then included as an integral part of the proposal which the government evaluates during the source selection.

Purpose

SOO expresses the basic, top-level objectives of the acquisition and is provided in the RFP in lieu of a government-written SOW. This approach gives the offerors the flexibility to develop cost effective solutions and the opportunity to propose innovative alternatives.

Approach

The government includes a brief (1- to 2-page) SOO in the RFP and requests that offerors provide a SOW in their proposal. The SOO is typically appended to section J of the RFP and does not become part of the contract. Instructions for the contractor prepared SOW would normally be included in or referenced by Section L.

SOO Development

Step 1: The RFP team develops a set of objectives compatible with the overall program direction including the following:

- User(s) operational requirements,
- Programmatic direction,
- Draft technical requirements, and
- Draft WBS and dictionary.

Step 2: Once the program objectives are defined, the SOO is constructed so that it addresses product-oriented goals and performance-oriented requirements.

SOO and Proposal Evaluations

Section L (Instructions to Offerors) of the RFP must include instructions to the offeror that require using the SOO to construct and submit a SOW. In Section M (Evaluation Criteria) the program office should include the criteria by which the proposals, including the contractor's draft SOW, will be evaluated. Because of its importance, the government's intention to evaluate the proposed SOW should be stressed in Sections L and M.

Offeror Development of the Statement of Work

The offeror should establish and define in clear, understandable terms:

- Non-specification requirements (the tasks that the contractor must do),
- What has to be delivered or provided in order for him to get paid,
- What data is necessary to support the effort, and
- Information that would show how the offerors would perform the work that could differentiate between them in proposal evaluation and contractor selection.

SOO Example:
Joint Air-to-Surface Standoff Missile (JASSM)
Statement of Objectives

The Air Force and Navy warfighters need a standoff missile that will destroy the enemies' war-sustaining capabilities with a launch standoff range outside the range of enemy area defenses. Offerors shall use the following objectives for the pre-EMD and EMD acquisition phases of the JASSM program along with other applicable portions of the RFP when preparing proposals and program plans. IMP events shall be traceable to this statement of objectives:

Pre-EMD Objectives

a. Demonstrate, at the sub-system level as a minimum, end-to-end performance of the system concept. Performance will be at the contractor-developed System Performance Specification requirements level determined during this phase without violation of any key performance parameters.

b. Demonstrate the ability to deliver an affordable and producible system at or under the average unit procurement price (AUPP).

c. Provide a JASSM system review including final system design, technical accomplishments, remaining technical risks and major tasks to be accomplished in EMD.

EMD Objectives

a. Demonstrate through test and/or analysis that all requirements as stated in the contractor generated System Performance Specification, derived from Operational Requirements, are met, including military utility (operational effectiveness and suitability).

b. Demonstrate ability to deliver an affordable and producible system at or under the AUPP requirement.

c. Demonstrate all production processes.

d. Produce production representative systems for operational test and evaluation, including combined development/operational test and evaluation.

At contract award the SOW, as changed through negotiations, becomes part of the contract and the standard for measuring contractor's effectiveness.

SUPPLEMENT 19-B

STATEMENT OF WORK (SOW)

The SOW is a specific statement of the work to be performed by the contractor. It is derived from the Program WBS (System Architecture). It should contain, at a minimum, a statement of scope and intent, as well as a logical and clear definition of all tasks required. The SOW normally consists of three parts:

Section 1: Scope – Defines overall purpose of the program and to what the SOW applies.

Section 2: Applicable Documents – Lists the specifications and standards referenced in Section 3.

Section 3: Requirements – States the tasks the contractor has to perform to provide the deliverables. Tasks should track with the WBS. The SOW describes tasks the contractor has to do. The specifications describe the products.

Statement of Work Preparation and Evaluation Strategies

SOWs should be written by an integrated team of competent and experienced members. The team should:

- Review and use the appropriate WBS for the SOW framework,

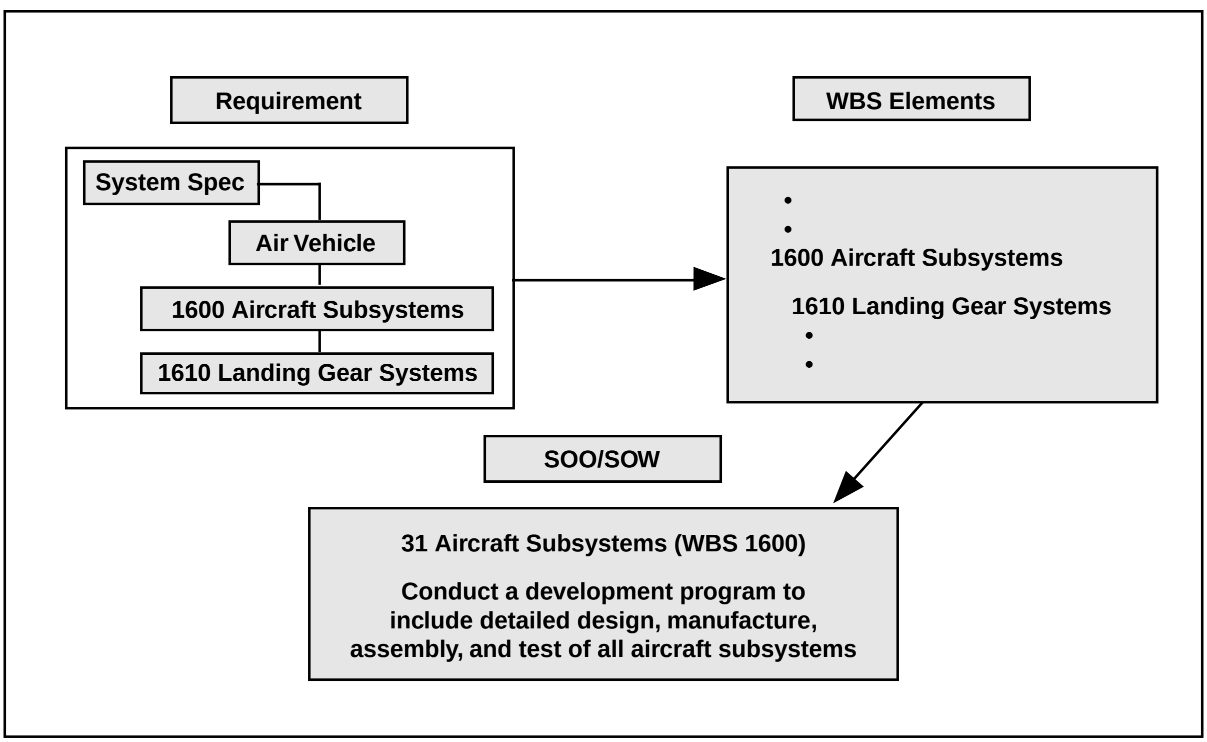


Figure 19-4. Requirement-WBS-SOW Flow

- Set SOW objectives in accordance with the Acquisition Plan and systems engineering planning,
- Develop a SOW tasking outline and check list,
- Establish schedule and deadlines, and
- Develop a comprehensive SOW from the above.

Performance-based SOW

The term *performance-based SOW* has become a common expression that relates to a SOW that tasks the contractor to perform the duties necessary to provide the required deliverables, but is not specific as to the process details. Basically, all SOWs should be performance based, however, past DoD generated SOWs have had the reputation of being overly directive. A properly developed SOW tasks the contractor without telling him how to accomplish the task.

Evaluating the SOW

The WBS facilitates a logical arrangement of the elements of the SOW and a tracing of work effort expended under each of the WBS elements. It helps integrated teams to ensure all requirements have been included, and provides a foundation for tracking program evolution and controlling the change process. As shown by Figure 19-4, the WBS serves as a link between the requirements and the SOW.

In the past, DoD usually wrote the SOW and, over time, an informal set of rules had been developed to assist in drafting them. While the government today generally does not write the SOW, but, rather, more often evaluates the contractor's proposed SOW, those same rules can assist in the government role of evaluator.

Statement of Work Rules

In section 1. ***Scope:***

DO NOT:

- Include directed work statements.
- Include data requirements or deliverable products.

In section 2. ***Applicable Documents:***

DO NOT:

- Include guidance documents that apply only to Government PMOs (e.g., DoD 5000 series and service regulations).

In section 3. ***Requirements:***

DO NOT:

- Define work tasks in terms of data to be delivered.
- Order, describe, or discuss CDRL data (OK to reference).
- Express work tasks in data terms.
- Invoke, cite, or discuss a DID.
- Invoke handbooks, service regulations, technical orders, or any other document not specifically written in accordance with MIL-STD-961/962.
- Specify how task is to be accomplished.
- Use the SOW to amend contract specifications.
- Specify technical proposal or performance criteria or evaluation factors.
- Establish delivery schedules.
- Over specify.

In section 3. ***Requirements:***

DO:

- Specify work requirements to be performed under contract.

- Set SOW objectives to reflect the acquisition plan and systems engineering planning.
- Provide a priceable set of tasks.
- Express work to be accomplished in work words.
- Use “shall” whenever a task is mandatory.
- Use “will” only to express a declaration of purpose or simple futurity.
- Use WBS as an outline.
- List tasks in chronological order.
- Limit paragraph numbering to 3rd sub-level (3.3.1.1.) – Protect Government interests.
- Allow for contractor’s creative effort.

SUPPLEMENT 19-C

CONTRACT DATA REQUIREMENTS LIST

The Contract Data Requirements List (CDRL) is a list of authorized data requirements for a specific procurement that forms a part of the contract. It is comprised of a series of DD Forms 1423 (Individual CDRL forms) containing data requirements and delivery instructions. CDRLs should be linked directly to SOW tasks and managed by the program office data manager. A sample CDRL data requirement is shown in Figure 19-5.

Data requirements can also be identified in the contract via Special Contract Clauses (Federal Acquisition Clauses.) Data required by the FAR clauses are usually required and managed by the Contracting Officer.

CONTRACT DATA REQUIREMENTS LIST

ATCH NR: 3 **TO EXHIBIT:** **SYSTEM/ITEM: ATF DEM/VAL PHASE**
TO CONTRACT/PR: F33657-86-C-2085 **CATEGORY: X** **CONTRACTOR: LOCKHEED**

1) 3100	2) SOW 3.1 3)	6) ASD/TASE	10) ONE/R	12) 60DAC	14)

4) OT E62011	5) SOW 3.1	7) IT	8) D	9)	11)	13) SEE 16

14)

ASD/TASE	2/0
ASD/TASM	2/0
ASD/TASL	2/0
ACO	1/0

15) 7/0

16)
BLK 4: SEE APPENDIXES TO CDRL FOR DID.
THIS DID IS TAILORED AS FOLLOWS:
(1) CONTRACTOR FORMAT IS ACCEPTABLE.
(2) CHANGE PARAGRAPH 2a OF DID TO READ: "PROGRAM RISK ANALYSIS. THIS SECTION SHALL DESCRIBE THE PLAN AND METHODOLOGY FOR A CONTINUING ASSESSMENT OF TECHNICAL, SUPPORTABILITY, COST, AND SCHEDULE RISKS OF THE SYSTEM PROGRAM. THIS SECTION SHOULD BE CONSISTENT WITH AND NOT DUPLICATE THE SYSTEM INTEGRATION PLAN (REFERENCE DI-S-3563/T); i.e., ONE PLAN MAY REFERENCE THE OTHER."
BLK 13: REVISIONS SHALL BE SUBMITTED AS REQUIRED BY CHANGE RESULTING FROM THE SYSTEMS ENGINEERING PROCESS.
NOTE: SCHEDULES ASSOCIATED WITH THIS PLAN SHALL BE INTEGRATED WITH THE MASTER PROGRAM PLANNING SCHEDULE SUBMITTED ON MAGNETIC MEDIA IN ACCORDANCE WITH DI-A-3007/T.

PREPARED BY:	DATE:	APPROVED BY:	DATE:
	86 JUN 11		86 JUNE 11

DD FORM 1423 ADPE ADAPTATION SEP 81 (ASD/YYD)

Figure 19-5. CDRL Single Data Item Requirement Example

Data Requirement Sources

Standard Data Item Descriptions (DID) define data content, preparation instructions, format, intended use, and recommended distribution of data required of the contractor for delivery. The Acquisition Management Systems and Data Requirements Control List (AMSDL) identifies acquisition management systems, source documents, and standard DIDs. With acquisition reform the use of DIDs has declined, and data item requirements now are either tailored DIDs or a set of requirements specifically written for the particular RFP in formats agreeable to the contractor and the government.

DD Form 1423 Road Map

Block 1: Data Item Number – represents the CDRL sequence number.

Block 2: Title of Data Item – same as the title entered in item 1 of the DID (DD Form 1664).

Block 4: Authority (Data Acquisition Document Number) – same as item 2 of the DID form and will include a "/t" to indicate DID has been tailored.

Block 5: Contract Reference – identifies the DID authorized in block 4 and the applicable document and paragraph numbers in the SOW from which the data flows.

Block 6: Requiring Office – activity responsible for advising the technical adequacy of the data.

Block 7: Specific Requirements – may be needed for inspection/acceptance of data.

Block 8: Approval Code – if "A," it is a critical data item requiring specific, advanced, written approval prior to distribution of the final data item.

Block 9: Distribution Statement Required:

Category A is unlimited-release to the public.

Category B is limited-release to government agencies.

Category C limits release to government agencies and their contractors.

Category D is limited-release to DoD offices and their contractors.

Category E is for release to DoD components only.

Category F is released only as directed and normally classified.

Block 12: Date of First Submission – indicates year/month/day of first submission and identifies specific event or milestone data is required.

Block 13: Date of Subsequent Submission – if data is submitted more than once, subsequent dates will be identified.

Block 14: Distribution – identify each addressee and identify the number of copies to be received by each. Use office symbols, format of data to be delivered, command initials, etc.

Block 16: Remarks – explain only tailored features of the DID, any additional information for blocks 1-15, and any resubmittal schedule or special conditions for updating data submitted for government approval.

SUPPLEMENT 19-D

THE SOURCE SELECTION PLAN

Prior to solicitation issuance, a source selection plan should be prepared by the Program Manager (PM), reviewed by the Contracting Officer, and approved by the Source Selection Authority (SSA). A Source Selection Plan (SSP) generally consists of three parts:

- The first part describes the organization, membership, and responsibilities of the source selection team,
- The second part identifies the evaluation factors, and
- The last part establishes detailed procedures for the evaluation of proposals.

Source Selection Organization

The SSA is responsible for selecting the source whose proposal is most advantageous to the government. The Source Selection Advisory Council (SSAC) provides advice to the SSA based on the Source Selection Evaluation Board's (SSEB's) findings and the collective experience of SSAC members. The SSEB generates the information the SSA needs by performing a comprehensive evaluation of each offeror's proposal. A Technical Evaluation Review Team(s) evaluates the technical portion of the proposals to support the SSEB. The process flow is shown in Figure 19-6.

The PM is responsible for developing and implementing the acquisition strategy, preparing the SSP, and obtaining SSA approval of the plan before the formal solicitation is issued to industry. The System Engineer or technical manager supports the PM's efforts. The Contracting Officer is responsible for preparation of solicitations and contracts, any communications with potential offerors or offerors, consistency of the SSP with requirements of the Federal Acquisition Regulation (FAR) and DoD FAR Supplement (DFARS), and award of the contract.

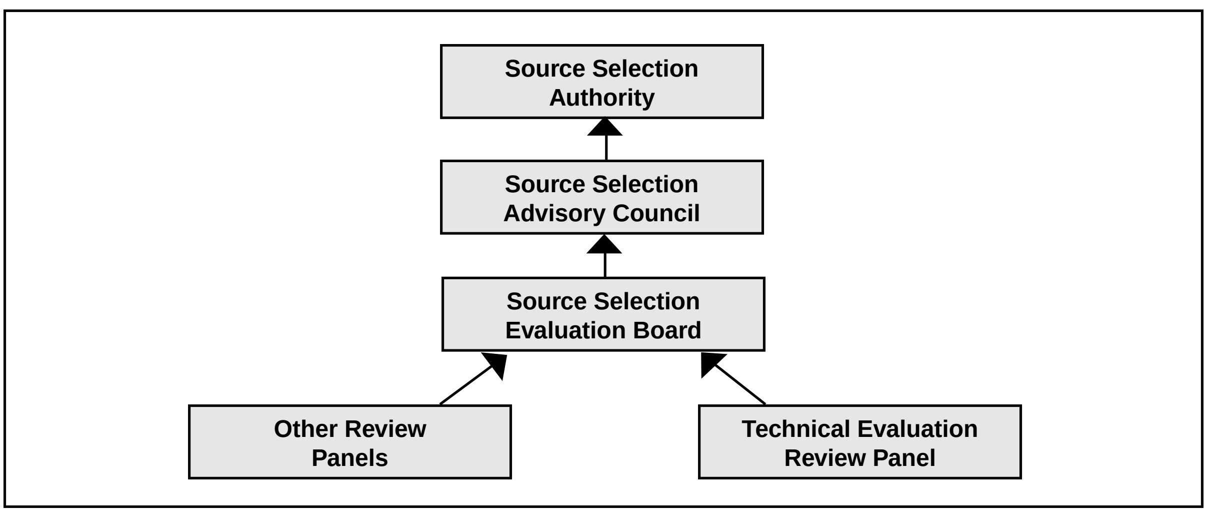


Figure 19-6. Source Selection Process

SSP Evaluation Factors

The evaluation factors are a list, in order of relative importance, of those aspects of a proposal that will be evaluated quantitatively and qualitatively to arrive at an integrated assessment as to which proposal can best meet the Government's need as described in the solicitation. Figure 19-7 shows an example of one evaluation category, life cycle cost. The purpose of the SSP evaluation is to inform offerors of the importance the Government attaches to various aspects of a proposal and to allow the government to make fair and reasoned differentiation between proposals.

In general the following guidance should be used in preparing evaluation factors:

- Limit the number of evaluation factors,

- Tailor the evaluation factors to the Government requirement (e.g., combined message of the SOO/SOW, specification, CDRL, etc.), and

- Cost is always an evaluation factor. The identification of the cost that is to be used and its relative importance in rating the proposal should be clearly identified.

Factors to Consider

There is not sufficient space here to attempt to exhaustively list all the factors that might influence the decision made in a source selection. The following are indicative of some of the key consideration, however:

- Is the supplier's proposal responsive to the government's needs as specified in the RFP?

- Is the supplier's proposal directly supportive of the system requirements specified in the system specification and SOO/SOW?

- Have the performance characteristics been adequately specified for the items proposed? Are they meaningful, *measurable*, and traceable from the system-level requirements?

- Have effectiveness factors been specified (e.g., reliability, maintainability, supportability, and availability?) Are they meaningful, *measurable*, and traceable, from the system-level requirements?

- Has the supplier addressed the requirement for test and evaluation of the proposed system element?

Rating (Points)	Evaluation Criteria – Life Cycle Cost
9-10	Offeror has included a complete Life Cycle Cost analysis that supports their proposal.
7-8	Offeror did not include a complete Life Cycle Cost analysis but has supported their design approach on the basis of Life Cycle Cost.
5-6	Offeror plans to complete a Life Cycle Cost analysis as part of the contract effort and has described the process that will be used.
3-4	Offeror plans to complete a Life Cycle Cost analysis as part of the contract effort but did not describe the process that will be used.
0-2	Life Cycle Cost was not addressed in the Offeror's proposal.

Figure 19-7. Evaluation Factors Example

- Have life cycle support requirements been identified (e.g., maintenance resource requirements, spare/repair parts, test and support equipment, personnel quantities and skills, etc?) Have these requirements been minimized to the extent possible through design?

- Does the proposed design configuration reflect growth potential or change flexibility?

- Has the supplier developed a comprehensive manufacturing and construction plan? Are key manufacturing processes identified along with their characteristics?

- Does the supplier have an adequate quality assurance and statistical process control programs?

- Does the supplier have a comprehensive planning effort (e.g., addresses program tasks, organizational structure and responsibilities, a WBS, task schedules, program monitoring and control procedures, etc.)?

- Does the supplier's proposal address all aspects of total life cycle cost?

- Does the supplier have previous experience in the design, development, and production of system elements/components which are similar in nature to the item proposed?

Proposal Evaluation

Proposal evaluation factors can be analyzed with any reasonable trade study approach. Figure 19-8 shows a common approach. In this approach each factor is rated based on the evaluation factor matrix established for each criteria, such as that shown in Figure 19-7. It is then multiplied by a weighting factor based on the perceived priority of each criteria. All the weighted evaluations are added together and the highest score wins.

Like trade studies the process should be examined for sensitivity problems; however, in the case of source selection, the check must be done with anticipated values prior to release of the RFP.

Evaluation Criteria	WT. Factor (%)	Proposal A		Proposal B		Proposal C	
		Rating	Score	Rating	Score	Rating	Score
A. Technical Requirements:	25						
1. Performance Characteristics	6	4	24	5	30	5	30
2. Effectiveness Factors	4	3	12	4	16	3	12
3. Design Approach	3	2	6	3	9	1	3
4. Design Documentation	4	3	12	4	16	2	8
5. Test and Evaluation Approach	2	2	4	1	2	2	4
6. Product Support Requirements	4	2	8	3	12	2	8
B. Production Capability	20						
1. Production Layout	8	5	40	6	48	6	48
2. Manufacturing Process	5	2	10	3	15	4	20
3. Quality Control Assurance	7	5	35	6	42	4	28
C. Management	20						
1. Planning (Plans/Schedules)	6	4	24	5	30	4	24
2. Organization Structure	4	4	16	4	12	4	16
3. Available Personnel Resources	5	3	15	3	20	3	15
4. Management Controls	5	3	15	3	20	4	20
D. Total Cost	25						
1. Acquisition Price	10	7	70	5	50	6	60
2. Life Cycle Cost	15	9	135	10	150	8	120
E. Additional Factors	10						
1. Prior Experience	4	4	16	3	12	3	12
2. Past Performance	6	5	30	5	30	3	18
Grand Total	100		476		516*		450

* Select Proposal B

Figure 19-8. Source Evaluation

CHAPTER 20

MANAGEMENT CONSIDERATIONS AND SUMMARY

20.1 MANAGEMENT CONSIDERATIONS

The Acquisition Reform Environment

No one involved in systems acquisition, either within the department or as a supplier, can avoid considering how to manage acquisition in the current reform environment. In many ways, rethinking the way we manage the systems engineering process is *implicit* in reforming acquisition management. Using performance specifications (instead of detailed design specifications), leaving design decisions in the hands of contractors, delaying government control of configuration baselines—all are reform measures related directly to systems engineering management. This text has already addressed and acknowledged managing the technical effort in a reform environment.

To a significant extent, the systems engineering processes—and systems engineers in general—are victims of their own successes in this environment. The systems engineering process was created and evolved to bring discipline to the business of producing very complex systems. It is intended to ensure that requirements are carefully analyzed, and that they flow down to detailed designs. The process demands that details are understood and managed. And the process has been successful. Since the 1960s manufacturers, in concert with government program offices, have produced a series of ever-increasingly capable and reliable systems using the processes described in this text. The problem is, in too many cases, we have overlaid the process with ever-increasing levels of controls, reports, and reviews. The result is that the cycle time required to produce systems has increased to unacceptable levels, even as technology life cycles have decreased precipitously. The fact is that, in too many cases, we are producing excellent systems, but systems that take too long to produce, cost too much, and are often outdated when they are finally produced. The demand for change has been sounded, and systems engineering management must respond if change is to take place. The question then becomes how should one manage to be successful in this environment? We have a process that produces good systems; how should we change the process that has served us well so that it serves us better?

At the heart of acquisition reform is this idea: we can improve our ability to provide our users with highly capable systems at reasonable cost and schedule. We can if we manage design and development in a way that takes full advantage of the expertise resident both with the government and the contractor. This translates into the government stating its needs in terms of performance outcomes desired, rather than in terms of specific design solutions required; and, likewise, in having contractors select detailed design approaches that deliver the performance demanded, and then taking responsibility for the performance actually achieved.

This approach has been implemented in DoD, and in other government agencies as well. In its earlier implementations, several cases occurred where the government managers, in an attempt to ensure that the government did not impose design solutions on contractors, chose to deliberately distance the government technical staff from contractors. This presumed that the contractor would step forward to ensure that necessary engineering disciplines and functions were covered. In more than one case, the evidence after the fact was that, as the government stepped back to a less directive role in design and development, the contractor did not take a corresponding step forward to ensure that normal engineering management disciplines were included. In several cases where problems arose, after-the-fact investigation showed important elements of the systems engineering process were either deliberately ignored or overlooked.

The problem in each case seems to have been failure to communicate expectations between the government and the contractor, compounded by a failure on the part of the government to ensure that normal engineering management disciplines were exercised. One of the more important lessons learned has been that while the systems engineering process can—and should be—tailored to the specific needs of the program, there is substantial risk ignoring elements of the process. Before one decides to skip phases, eliminate reviews, or take other actions that appear to deliver shortened schedules and less cost, one must ensure that those decisions are appropriate for the risks that characterize the program.

Arbitrary engineering management decisions yield poor technical results. One of the primary requirements inherent in systems engineering is to assess the engineering management program for its consistency with the technical realities and risks confronted, and to communicate his/her findings and recommendations to management. DoD policy is quite clear on this issue. The government is not, in most cases, expected to take the lead in the development of design solutions. That, however, does not relieve the government of its responsibilities to the taxpayers to ensure that sound technical and management processes are in place. The systems engineer must take the lead role in establishing the technical management requirements for the program and seeing that those requirements are communicated clearly to program managers and to the contractor.

Communication – Trust and Integrity

Clearly, one of the fundamental requirements for an effective systems engineer is the ability to communicate. Key to effective communication is the rudimentary understanding that communication involves two elements—a transmitter and a receiver. Even if we have a valid message and the capacity for expressing our positions in terms that enable others to understand what we are saying, true communication may not take place if the intended receiver chooses not to receive our message. What can we do, as engineering managers to help our own cause as far as ensuring that our communications are received and understood?

Much can be done to condition others to listen and give serious consideration to what one says, and, of course, the opposite is equally true—one can condition others to ignore what he/she says. It is primarily a matter of establishing credibility based on integrity and trust.

First, however, it is appropriate to discuss the systems engineer's role as a member of the management team. Systems engineering, as practiced in DoD, is fundamentally the practice of engineering management. The systems engineer is expected to integrate not only the technical disciplines in reaching recommendations, but also to integrate traditional management concerns such as cost, schedule, and policy into the technical management equation. In this role, senior levels of management expect the systems engineer to understand the policies that govern the program, and to appreciate the imperatives of cost and schedule. Furthermore, in the absence of compelling reasons to the contrary, they expect support of the policies enunciated and they expect the senior engineer to balance technical performance objectives with cost and schedule constraints.

Does this mean that the engineer should place his obligation to be a supportive team member above his ethical obligation to provide honest engineering judgment? Absolutely not! But it does mean that, if one is to gain a fair hearing for expression of reservations based on engineering judgment, one must be viewed as a member of the team. The individual who always fights the system, always objects to established policy, and, in general, refuses to try to see other points of view will eventually become isolated. When others cease listening, the communication stops and even valid points of view are lost because the intended audience is no longer receiving the message—valid or not.

In addition to being team players, engineering managers can further condition others to be receptive to their views by establishing a reputation for making reasoned judgments. A primary requirement for establishing such a reputation is that managers must have technical expertise. They must be able to make technical judgments grounded in a sound understanding of the principles that govern science and technology. Systems engineers must have the education and the experience that justifies confidence in their technical judgments. In the absence of that kind of expertise, it is unlikely that engineering managers will be able to gain the respect of those with whom they must work. And yet, systems engineers cannot be expert in all the areas that must be integrated in order to create a successful system. Consequently, systems engineers must recognize the limits of their expertise and seek advice when those limits are reached. And, of course, systems engineers must have built a reputation for integrity. They must have demonstrated a willingness to make the principled stand when that is required and to make the tough call, even when there are substantial pressures to do otherwise.

Another, perhaps small way, that engineers can improve communication with other members of their teams (especially those without an engineering background) is to have confidence in the position being articulated and to articulate the position concisely. The natural tendency of many engineers is to put forward their position on a subject along with all the facts, figures, data and required proofs that resulted in the position being taken. This sometimes results in explaining how a watch works when all that was asked was "What time is it?" Unless demonstrated otherwise, team members will generally trust the engineer's judgment and will assume that all the required rationale is in place, without having to see it. There are some times when it is appropriate to describe how the watch works, but many times communication is enhanced and time saved by providing a confident and concise answer.

When systems engineers show themselves to be strong and knowledgeable, able to operate effectively in a team environment, then communication problems are unlikely to stand in the way of effective engineering management.

20.2 ETHICAL CONSIDERATIONS

The practice of engineering exists in an environment of many competing interests. Cost and schedule pressures; changes in operational threats, requirements, technology, laws, and policies; and changes in the emphasis on tailoring policies in a common-sense way are a few examples. These competing interests are exposed on a daily basis as organizations embrace the integrated product and process development approach. The communication techniques described earlier in this chapter, and the systems engineering tools described in earlier chapters of this book, provide guidance for engineers in effectively advocating the importance of the technical aspects of the product in this environment of competing interests.

But, what do engineers do when, in their opinion, the integrated team or its leadership are not putting adequate emphasis on the technical issues? This question becomes especially difficult in the cases of product safety or when human life is at stake. There is no explicit set of rules that directs the individual in handling issues of ethical integrity. Ethics is the responsibility of everyone on the integrated team. Engineers, while clearly the advocate for the technical aspects of the intgrated solution, do not have a special role as ethical watchdogs because of their technical knowledge.

Richard T. De George in his article entitled *Ethical Responsibilities of Engineers in Large Organizations: The Pinto Case*[1] makes the following case: "The myth that ethics has no place in engineering

[1] *Ethical Issues in Engineering*, Johnson, Ch 15.

has been attacked, and at least in some corners of the engineering profession been put to rest. Another myth, however, is emerging to take its place—the myth of the engineer as moral hero."

This emphasis, De George believes, is misplaced. "The zeal of some preachers, however, has gone too far, piling moral responsibility upon moral responsibility on the shoulders of the engineer. Though engineers are members of a profession that holds public safety paramount, we cannot reasonably expect engineers to be willing to sacrifice their jobs each day for principle and to have a whistle ever by their sides ready to blow if their firm strays from what they perceive to be the morally right course of action."

What then is the responsibility of engineers to speak out? De George suggests as a rule of thumb that engineers and others in a large organization are morally permitted to go public with information about the safety of a product if the following conditions are met:

1. If the harm that will be done by the product to the public is serious and considerable.

2. If they make their concerns known to their superiors.

3. If, getting no satisfaction from their immediate supervisors, they exhaust the channels available within the operation, including going to the board of directors (or equivalent).

De George believes if they still get no action at this point, engineers or others are morally permitted to make their concerns public but not morally obligated to do so. To have a moral obligation to go public he adds two additional conditions to those above:

4. The person must have documented evidence that would convince a reasonable, impartial observer that his/her view of the situation is correct and the company policy wrong.

5. There must be strong evidence that making the information public will in fact prevent the threatened serious harm.

Most ethical dilemmas in engineering management can be traced to different objectives and expectations in the vertical chain of command. Higher authority knows the external pressures that impact programs and tends to focus on them. System engineers know the realities of the on-going development process and tend to focus on the internal technical process. Unless there is communication between the two, misunderstandings and late information can generate reactive decisions and potential ethical dilemmas. The challenge for system engineers is to improve communication to help unify objectives and expectations. Divisive ethical issues can be avoided where communication is respected and maintained.

20.3 SUMMARY

The material presented in this book is focused on the details of the classic systems engineering process and the role of the systems engineer as the primary practitioner where the activities included in that process are concerned. The systems engineering process described has been used successfully in both DoD and commercial product development for decades. In that sense, little new or revolutionary material has been introduced in this text. Rather, we have tried to describe this time-proven process at a level of detail that makes it logical and understandable as a tool to use to plan, design, and develop products that must meet a defined set of requirements.

In DoD, systems engineers must assume roles of engineering managers on the program or project assigned. They must understand that the role of the systems engineer is necessarily different from that normal to the narrowly specialized functional engineer, yet it is also different from the role played by the program manager. In a sense, the role of the systems engineer is a delicate one, striving to balance technical concerns with the real management pressures deriving from cost, schedule, and policy. The systems engineer is often the person in the middle; it is seldom a comfortable position. This text has been aimed at that individual.

The first two parts of the text were intended to first give the reader a comprehensive overview of systems engineering as a practice and to demonstrate the role that systems engineering plays within the DoD acquisition management process. Part 2, in particular, was intended to provide relatively detailed insights into the specific activities that make up the process. The government systems engineer may find him/herself deeply involved in some of the detailed activities that are included in the process, while less involved in others. For example, government systems engineers may find themselves very involved in requirements definition and analysis, but less directly involved in design synthesis. However, the fact that government engineers do not directly synthesize designs does not relieve them from a responsibility to understand the process and to ensure that sound practices are pursued in reaching design decisions. It is for this reason that understanding details of the process are critical.

Part 3 of the book is perhaps the heart of the text from an engineering management perspective. In Part 3, we have presented discussions on a series of topics under the general heading of Systems Analysis and Control. The engine that translates requirements into designs is defined by the requirements analysis, functional analysis and allocation, and design synthesis sequence of activities. Much of the role of the systems engineer is to evaluate progress, consider alternatives, and ensure the product remains consistent and true to the requirements upon which the design is based. The tools and techniques presented in Part 3 are the primary means by which a good engineering management effort accomplishes these tasks.

Finally, in Part 4, we presented some of the considerations beyond the implementation of a disciplined systems engineering process that the engineering manager must consider in order to be successful. Particularly in today's environment where new starts are few and resources often limited, the planning function and the issues associated with product improvement and integrated team management must move to the forefront of the systems engineer's thinking from the very early stages of work on any system.

This book has attempted to summarize the primary activities and issues associated with the conduct and management of technical activities on DoD programs and projects. It was written to supplement the material presented courses at the Defense Systems Management College. The disciplined application of the principles associated with systems engineering has been recognized as one indicator of likely success in complex programs. As always, however, the key is for the practitioner to be able to absorb these fundamental principles and then to tailor them to the specific circumstances confronted. We hope that the book will prove useful in the future challenges that readers will face as engineering managers.

GLOSSARY

GLOSSARY

SYSTEMS ENGINEERING FUNDAMENTALS

AAAV Advanced Amphibious Assault Vehicle

ACAT Acquisition Category

ACR Alternative Concept Review

AMSDL Acquisition Management Systems Data List

ASR Alternative Systems Review

AUPP Average Unit Procurement Price

AWP Awaiting Parts

BL Baseline

BLRIP Beyond Low Rate Initial Production

C4ISR Command, ontrol, Communications, Computers, Intelligence, and Reconnaissance

CAD Computer-Aided Design

CAE Computer-Aided Engineering

CAIV Cost As an Independent Variable

CALS Continuous Acquisition and Life Cycle Support

CAM Computer-Aided Manufacturing

CASE Computer-Aided Systems Engineering

CATIA Computer-Aided Three-Dimensional Interactive Application

CCB Configuration Control Board

CCR Contract Change Request

CDR Critical Design Review

CDRL Contract Data Requirement List

CDS Concept Design Sheet

CE Concept Exploration

CEO	Chief Executive Officer
CI	Configuration Item
Circular A-109	Major Systems Acquisitions
CM	Configuration Management
CM	Control Manager
COTS	Commercial Off-The-Shelf
CSCI	Computer Software Configuration Item
CWI	Continuous Wave Illumination
DAU	Defense Acquisition University
DCMC	Defense Contract Management Command
DDR	Detail Design Review
DFARS	Defense Supplement to the Federal Acquisition Regulation
DID	Data Item Description
DoD	Department of Defense
DoD 5000.2-R	Mandatory Procedures for Major Defense Acquisition Programs (MDAPs), and Major Automated Information System Acquisition Programs (MAIS)
DoDISS	DoD Index of Specifications and Standards
DSMC	Defense Systems Management College
DT	Developmental Testing
DTC	Design To Cost
DT&E	Developmental Test and Evaluation
EC	Engineering Change
ECP	Engineering Change Proposal
EDI	Electronic Data Interchange
EIA	Electronic Industries Alliance
EIA IS 632	Electronic Industries Association Interim Standard 632, on Systems Engineering
EIA IS-649	Electronic Industries Association Interim Standard 649, on Configuration Management
EOA	Early Operational Assessments

FAR Federal Acquisition Regulation

FCA Functional Configuration Audit

FEO Field Engineering Order

FFBD Functional Flow Block Diagram

FIPS Federal Information Processing Standard

FMECA Failure Modes, Effects, and Criticality Analysis

FOT&E Follow-On Operational Test and Evaluation

FQR Formal Qualification Review

GFE Government Furnished Equipment

GFM Government Furnished Material

ICD Interface Control Documentation

ICWG Interface Control Working Group

IDE Integrated Digital Environment

IDEF Integration Definition Function

IDEF0 Integrated Definition for Function Modeling

IDEF1x Integration Definition for Information Modeling

IEEE Institute of Electrical and Electronics Engineers

IEEE/EIA 12207 IEEE/EIA Standard 12207, Software Life Cycle Processes

IEEE P1220 IEEE Draft Standard 1220, Application and Management of the Systems Engineering Process

IFB Invitation for Bid

IIPT Integrating Integrated Product Teams

IMS Integrated Master Schedule

IOC Initial Operational Capability

IOT&E Initial Operational Test and Evaluation

IPPD Integrated Product and Process Development

IPR In-Progress/Process Review

IPT Integrated Product Teams

JASSM Joint Air-to-Surface Standoff Missile

JROC Joint Requirements Oversight Council

JTA Joint Technical Architecture

KPPs Key Performance Parameters

LFT&E Live Fire Test and Evaluation

LRU Line-Replaceable Unit

LRIP Low Rate Initial Production

M&S Modeling and Stimulation

MAIS Major Automated Information System

MAISRC Major Automated Information Systems Review Council

MBTF Mean Time Between Failure

MDA Milestone Decision Authority

MDAP Major Defense Acquisition Program

MIL-HDBK-61 Military Handbook 61, on Configuration Management

MIL-HDBK-881 Military Handbook 881, on Work Breakdown Structure

MIL-STD 499A Military Standard 499A, on Engineering Management

MIL-STD-961D Military Standard 961D, on Standard Practice for Defense Specifications

MIL-STD 962 Military Standard 962, on Format and Content of Defense Standards

MIL-STD-973 Military Standard 973, on Configuration Management

MNS Mission Need Statement

MOE Measure of Effectiveness

MOP Measure of Performance

MOS Measure of Suitability

MRP II Manufacturing Resource Planning II

MS Milestone

MTTR Mean Time To Repair

NDI Non-Developmental Item

NIST National Institute of Standards and Technology

NRTS Not Repairable This Station

OA Operational Assessment
OIPT Overarching Integrated Product Teams
OMB Office of Management and Budget
OPS Operations
ORD Operational Requirements Document
OSD Office of the Secretary of Defense
OT&E Operational Test and Evaluation

P3I Preplanned Product Improvement
PAR Production Approval Reviews
PCA Physical Configuration Audit
PDR Preliminary Design Review
PDRR Program Definition and Risk Reduction
PEO Program Executive Office
PM Program Manager
PME Program/Project Manager – Electronics
PMO Program Management Office
PMT Program Management Team
PPBS Planning, Programming and Budgeting System
PRR Production Readiness Review

QA Quality Assurance
QFD Quality Function Deployment

R&D Research and Development
RAS Requirements Allocation Sheets
RCS Radar Cross Section
RDT&E Research, Development, Test and Evaluation
RFP Request for Proposal

S&T Science and Technology

SBA Simulation Based Acquisition

SBD Schematic Block Diagram

SD&E System Development and Demonstration

SDefR System Definition Review (as referred to in IEEE P1220)

SDR System Design Review

SE Systems Engineering

Section L Instructions to Offerors (Portion of Uniform Contract Format)

Section M Evaluation Criteria (Portion of Uniform Contract Format)

SEDS Systems Engineering Detail Schedule

SEMS Systems Engineering Master Schedule

SEP Systems Engineering Process

SFR System Functional Review

SI Software Item

SI&T System Integration and Test

SOO Statement of Objectives

SOW Statement of Work

SPEC Specification

SSA Source Selection Authority

SSAC Source Selection Advisory Council

SSEB Source Selection Evaluation Board

SSP Source Selection Plan

SSR Software Specification Review

SRR System Requirements Review

SRU Shop-Replaceable Unit

STD Standard

SVR System Verification Review

S/W Software

T&E Test and Evaluation

TDP Technical Data Package

TEMP Test and Evaluation Master Plan

TLS Timeline Analysis Sheet

TOC Team Operating Contract

TPM Technical Performance Measurement

TPWG Test Planning Work Group

TRR Test Readiness Review

VV&A Verfication, Validation, and Accreditation

WIPT Working-Level Integrated Product Team