ANSI/EIA-632-1998
Approved: January 7, 1999

EIA STANDARD

EIA-632

Processes for Engineering a System

EIA-632

(Upgrade and Revision of EIA/IS-632)

JANUARY 1999

ELECTRONIC INDUSTRIES ALLIANCE

GOVERNMENT ELECTRONICS AND INFORMATION TECHNOLOGY ASSOCIATION ENGINEERING DEPARTMENT

GEIA
Government Electronics and Information Technology Association

A SECTOR OF

EIA
Electronic Industries Alliance

NOTICE

EIA Engineering Standards and Publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need. Existence of such Standards and Publications shall not in any respect preclude any member or nonmember of EIA from manufacturing or selling products not conforming to such Standards and Publications, nor shall the existence of such Standards and Publications preclude their voluntary use by those other than EIA members, whether the standard is to be used either domestically or internationally.

Standards and Publications are adopted by EIA in accordance with the American National Standards Institute (ANSI) patent policy. By such action, EIA does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the Standard or Publication.

This EIA Standard is considered to have International Standardization implication, but the International Electrotechnical Commission activity has not progressed to the point where a valid comparison between the EIA Standard and the IEC document can be made.

This Standard does not purport to address all safety problems associated with its use or all applicable regulatory requirements. It is the responsibility of the user of this Standard to establish appropriate safety and health practices and to determine the applicability of regulatory limitations before its use.

(From Standards Proposal No. 3537-A, formulated under the cognizance of the EIA 632 Working Group of the G-47 Systems Engineering Committee.)

Published by

ELECTRONIC INDUSTRIES ALLIANCE 1998
Engineering Department
2500 Wilson Boulevard
Arlington, VA 22201

PRICE: Please refer to the current
Catalog of EIA, JEDEC, and TIA STANDARDS and ENGINEERING PUBLICATIONS
or call Global Engineering Documents, USA and Canada (1-800-854-7179)
International (303-397-7956)

Processes for Engineering a System

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF REQUIREMENTS

THIS PAGE LEFT BLANK

Foreword

This Standard is intended to enable an enterprise to strengthen its competitiveness in global markets by engineering and producing quality systems, and by delivering its products on time at an affordable price or cost. The focus, therefore, is on conceptualizing, creating, and realizing a system and the products that make up a system.

This Standard was developed as a joint project of the Electronic Industries Alliance (EIA) and the International Council on Systems Engineering (INCOSE). This effort was chartered by the G-47 Systems Engineering Committee of EIA and has been designated as Project PN-3537. This Standard has been approved by the EIA Engineering Department Executive Committee.

Intended uses of this Standard include

a) benchmarking by an enterprise against the requirements of this Standard for engineering a system, or portion thereof;

b) preparing enterprise standards, policies, and procedures for engineering a system;

c) developing lower-tier industry- or domain-specific process standards;

d) developing process capability and assessment models;

e) establishing terminology and concepts for better communications;

f) developing training and education curricula;

g) preparing plans for actual development of a product.

Use is not limited to specific disciplines, industry sectors, or technology domains.

To provide each enterprise with the greatest degree of flexibility for adapting to changing environments while maintaining the integrity of adopted processes, this Standard

a) limits the set of required processes to those directly related to the technical aspects of engineering systems;

b) defines representative tasks associated with each process;

c) includes the relevant information flows and interactions with enterprise and project entities.

This Standard is intended to define "what to do" with respect to the processes for engineering a system. ANSI/EIA-731, *Systems Engineering Capability*, provides a capability model and assessment method as a basis and means for determining "how well" the processes in ANSI/EIA-632 are defined and implemented.

This Standard is consistent with ISO 9000 in that it provides processes that can be adopted by enterprises for engineering systems.

Annex A is normative. Annexes B through G are informative.

THIS PAGE LEFT BLANK

Introduction

Background

In June 1994, a working group of industry associations, the International Council on Systems Engineering (INCOSE), and the Department of Defense developed an interim standard for the engineering of systems. This effort was led by the G-47 Committee on Systems Engineering of the Electronic Industries Alliance (EIA). EIA/IS 632 was intended to provide a standard for use by commercial enterprises, as well as government agencies and their development contractors.

In April 1995, a formal working group was established under Project PN-3537 and with EIA and INCOSE sponsorship to generate and release this full Standard. The joint working group decided that it would best serve U. S. industry to develop a "top-tier" standard applicable across all industry sectors and technology domains. As a result, the contents of this Standard are an abstraction of the essential features of the engineering practices described in the interim version of this Standard.

Contents of this Standard

This Standard defines a systematic approach to engineering or reengineering a system, incorporating best practices that have evolved during the second half of the twentieth century. The defined approach has three premises:

a) A system is one or more end products and sets of related enabling products that allow end products, over their life cycle of use, to meet stakeholder needs and expectations;

b) Products are an integrated composite of hierarchical elements so integrated as to meet the defined stakeholder requirements;

c) The engineering of a system and its related products is accomplished by applying a set of processes to each element of the system hierarchy by a multidisciplinary team of people who have the requisite knowledge and skills.

The systematic approach of this Standard is applicable for: (1) completing corrective actions, (2) making refinements, (3) developing derivatives, (4) producing modifications, and (5) updating existing products, (6) creating and realizing new systems, and (7) allowing for the safe and cost-effective disposal (retirement) of a system. This approach is incrementally applied in an engineering life cycle framework that can be implemented during any one or more phases of an enterprise-based life cycle (for example, during production, operations, support, or disposal).

Voluntary Compliance

Adoption of this Standard is intended to be entirely voluntary, within the discretion of individual enterprises or other individual organizations.

Document Organization

The Standard is organized as follows:

Clause 1	*Scope*	states the purpose of the Standard and defines the particular processes to which it is intended to apply.
Clause 2	*Normative references*	lists other standards that are so referred to in the text as to make them indispensable in applying this Standard.
Clause 3	*Definitions and acronyms*	defines special use terms and acronyms.
Clause 4	*Requirements*	contains the requirements for the processes that are central to engineering a system. Representative tasks associated with each process are defined.
Clause 5	*Application context*	describes the context in which the processes of this Standard are applied.
Clause 6	*Application key concepts*	describes key concepts related to applying the processes of Clause 4 to generate and integrate the layers of end products and enabling products needed for engineering a system.
Annex A	*Glossary*	gives definitions for words that are used in a specific technical way in the body of the Standard. Only those terms for which the normal dictionary definition does not suffice are included.
Annex B	*Enterprise-based life cycle*	describes the management-life cycle phases in which a system, or portion thereof, is incrementally engineered.
Annex C	*Process task outcomes*	provides expected outcomes for the representative tasks identified in Clause 4.
Annex D	*Planning documents*	lists typical source, technical, and other documents related to engineering a system and their contents.
Annex E	*System technical reviews*	describes the necessary technical reviews for assessing progress against technical plans and requirements, and for assessing planned tasks.
Annex F	*Unprecedented and precedented development*	describes the application of the requirements of System Design processes for cases when system development is either unprecedented or precedented.
Annex G	*Requirement relationships*	defines different types of requirements and the relationship between these types and the logical and physical solution representations.

Acknowledgments

This document was developed by a team of subject matter experts (SME), including representatives from the following organizations:

—Electronic Industries Alliance (EIA)
—International Council on Systems Engineering (INCOSE).

Members of the Technical Committee are listed below:

James Martin, Raytheon Systems Co.
Chairman, Working Group and Technical Committee

Richard Harwell	Lockheed Martin Aeronautical Systems	INCOSE
Jerome (Jerry) Lake	Systems Management international	SME
John Velman	Hughes Space and Communications Co.	EIA

The Technical Committee was established to integrate and validate inputs from the Working Group and to disposition comments from Working Group members and from key reviewers. Other members of the Working Group are listed below:

Karl Arunski	Raytheon Systems Co.	EIA
Jim Johnson	Northrop Grumman, ESSD, Oceanic Systems	EIA
Ken LaSala	National Oceanic & Atmospheric Admin.	EIA
Ken Peterson	Boeing Defense and Space Co.	EIA
Cho Shimizu	Boeing Defense and Space Co.	EIA
Barbara Armistead	PRC, Inc.	INCOSE
Bruce Pittman	Profit Engineering Technologies	INCOSE
Richard Rountree	The Aerospace Corporation	INCOSE
Robert Shishko	CalTech/NASA Jet Propulsion Laboratory	INCOSE
John Snoderly	Defense Systems Management College	INCOSE
Randy Zittel	Defense Systems Management College	INCOSE
James Armstrong	Software Productivity Consortium	SME
Steve Tavan	Draper Laboratory	SME

Two liaison members involved with ISO standardization activities related to this Standard participated in working group activities:

Stuart Arnold	Defence Evaluation Research Agency	ISO Liaison
Raghu Singh	Federal Aviation Administration	ISO Liaison

THIS PAGE LEFT BLANK

1 Scope

1.1 Purpose

The purpose of this Standard is to provide an integrated set of fundamental processes to aid a developer in the engineering or reengineering of a system.

Use of this Standard is intended to help developers

a) establish and evolve a complete and consistent set of requirements that will enable delivery of feasible and cost-effective system solutions;

b) satisfy requirements within cost, schedule, and risk constraints;

c) provide a system, or any portion of a system, that satisfies stakeholders over the life of the products that make up the system.

NOTE—The term *product* is used in this standard to mean: a *physical item*, such as a satellite (*end product*), or any of its component parts (*end products*); a software item such as a stand-alone application to run within an existing system *(end product*); or a *document* such as a plan, or a *service* such as test, training, or maintenance support, or *equipment* such as a simulator (enabling products).

d) provide for the safe and/or cost-effective disposal or retirement of a system.

1.2 Coverage

This Standard defines processes for engineering a system, as shown in Figure 1.2. These have been organized into five groups.

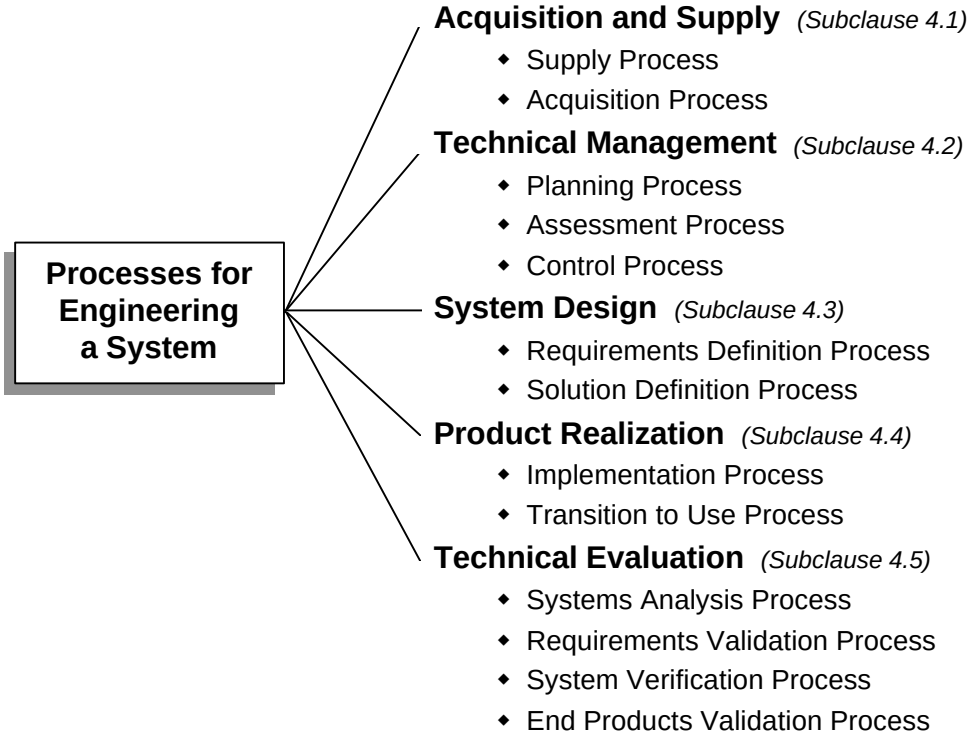

Figure 1.2—Fundamental processes for engineering a system

1.3 Intended applications

This Standard is intended to be applicable to the engineering or the reengineering of

a) commercial or non-commercial systems, or portions thereof;

b) any system, small or large, simple or complex, software-intensive or not, precedented or unprecedented;

c) systems containing products made up of hardware, software, firmware, personnel, facilities, data, materials, services, techniques, or processes (or combinations thereof);

d) a new system or a legacy system, or portions thereof.

> NOTE—The specific tasks and implementation methods for applying the processes required by this Standard can vary, for example, between commercial and government projects, or between customer-involved and internally developed projects. The fundamental processes, however, are the same in all cases.

The requirements of this Standard, or a designated subset, are intended to be applied by establishing enterprise policies and procedures that define the requirements for project implementation of the adopted processes of this Standard. The application of this Standard with respect to enterprises and projects is shown in Figure 1.3.

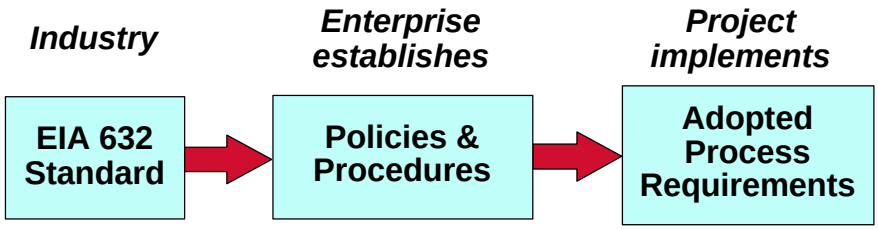

Figure 1.3—Application of this Standard

1.4 Limitations

This Standard does not specify the details of "how to" implement process requirements for engineering a system. Nor does it specify the methods or tools a developer would use to implement the process requirements. It is intended that the developer select or define methods and tools that are applicable to the development, and that are consistent with enterprise policies and procedures.

This Standard does not prescribe the name, format, content, structure, or medium of documentation.

2 Normative references

The following standards contain provisions that, through reference in this text, constitute provisions of this Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below.

ANSI/EIA-649-1998, *National Consensus Standard for Configuration Management, August 1998*

IEEE/EIA 12207.0-1996 (An American National Standard), Industry Implementation of International Standard ISO/IEC : 1995, *Software life cycle processes*

IEEE/EIA 12207.1-1997 (An American National Standard), Industry Implementation of International Standard ISO/IEC : 1995, *Software life cycle processes-Life cycle data*

IEEE/EIA 12207.2-1997 (An American National Standard), Industry Implementation of International Standard ISO/IEC : 1995, *Software life cycle processes-Implementation considerations*

3 Definitions and acronyms

3.1 Key terms

Definitions for special use of terms are contained in Annex A, Glossary.

3.2 Acronyms

Acronyms used in this Standard are shown below.

EIA	Electronic Industries Alliance
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council on Systems Engineering
IS	Interim Standard

NOTE—Acronyms that are spelled out in the text and not used elsewhere in the document are not listed above.

3.3 Terminology

The word ***shall*** identifies mandatory provisions of this Standard. The word ***should*** identifies recommended provisions of this Standard. The word ***may*** identifies permissive provisions of this Standard.

4 Requirements

This Clause provides requirements for processes used in engineering a system and is applicable to any product development regardless of its place in the hierarchy of the system structure (see Clause 6) or the enterprise-based life cycle phase (see Annex B). The processes are applicable for the engineering or reengineering of the end products that make up a system, as well as the development of enabling products required to provide life-cycle support to system end products. Figure 4a shows the relationships between the processes of this Standard.

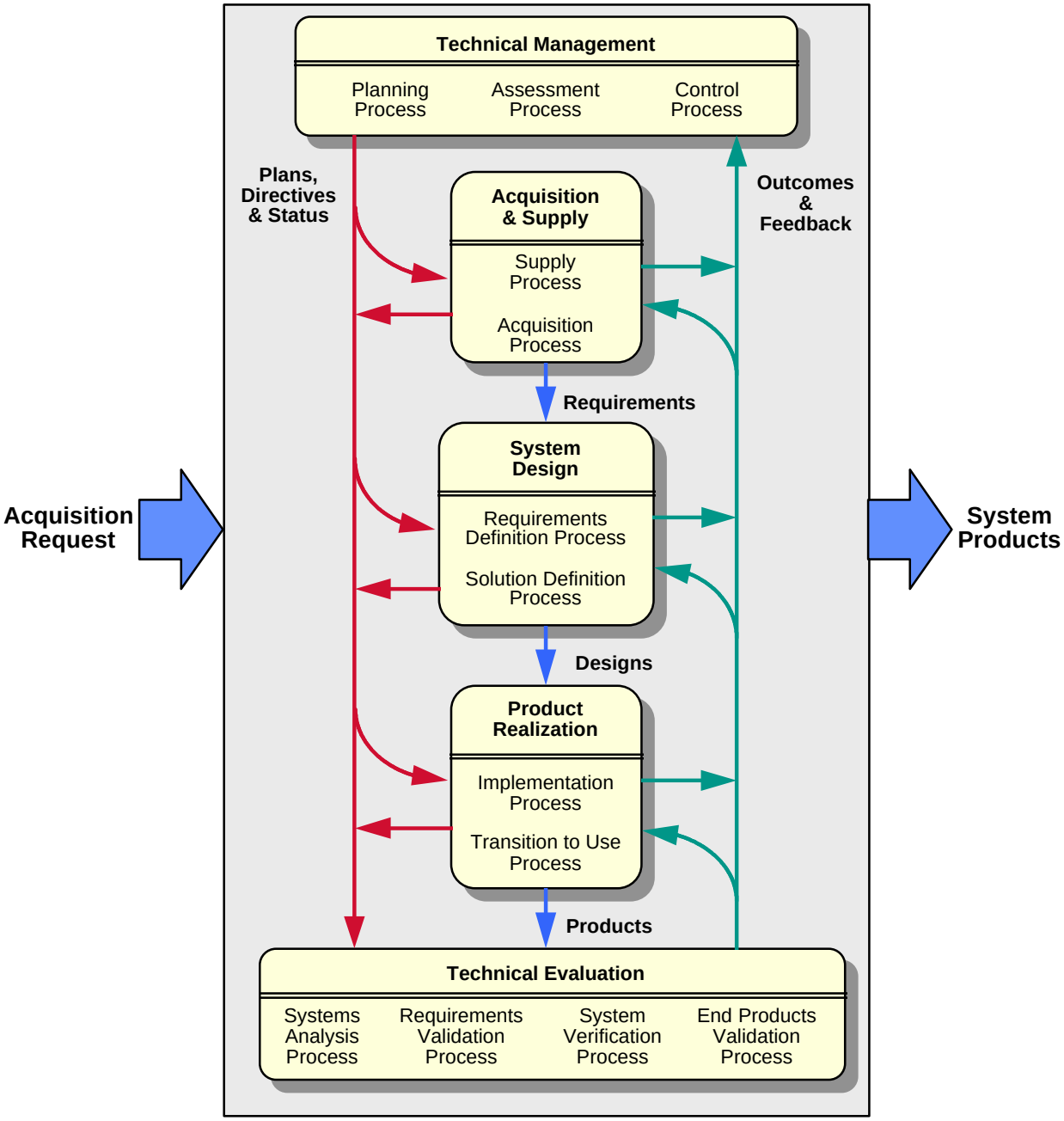

Figure 4a—Relationship of processes for engineering a system

NOTES

1 The application of the processes shown in Figure 4a is discussed in Clause 6 and Annex F. Appropriate processes of Figure 4a are applied recursively and iteratively to define the system products of the system hierarchy from the top down, and then, to implement and transition the system products from the bottom up to the user or customer.

2 Although the requirements in this Standard are presented sequentially, in practice many associated tasks are concurrent and highly iterative, and have interactive dependencies that lead to alteration of previously established technical requirements.

This Standard specifies the 33 requirements as shown in Figure 4b.

SUPPLY PROCESS REQUIREMENTS

1—Product Supply

ACQUISITION PROCESS REQUIREMENTS

2—Product Acquisition

3—Supplier Performance

PLANNING PROCESS REQUIREMENTS

4—Process Implementation Strategy

5—Technical Effort Definition

6—Schedule and Organization

7—Technical Plans

8—Work Directives

ASSESSMENT PROCESS REQUIREMENTS

9—Progress Against Plans and Schedules

10—Progress Against Requirements

11—Technical Reviews

CONTROL PROCESS REQUIREMENTS

12—Outcomes Management

13—Information Dissemination

REQUIREMENTS DEFINITION PROCESS REQUIREMENTS

14—Acquirer Requirements

15—Other Stakeholder Requirements

16—System Technical Requirements

SOLUTION DEFINITION PROCESS REQUIREMENTS

17—Logical Solution Representations

18—Physical Solution Representations

19—Specified Requirements

IMPLEMENTATION PROCESS REQUIREMENTS

20—Implementation

TRANSITION TO USE PROCESS REQUIREMENTS

21—Transition to Use

SYSTEMS ANALYSIS PROCESS REQUIREMENTS

22—Effectiveness Analysis

23—Tradeoff Analysis

24—Risk Analysis

REQUIREMENTS VALIDATION PROCESS REQUIREMENTS

25—Requirement Statements Validation

26—Acquirer Requirements Validation

27—Other Stakeholder Requirements Validation

28—System Technical Requirements Validation

29—Logical Solution Representations Validation

SYSTEM VERIFICATION PROCESS REQUIREMENTS

30—Design Solution Verification

31—End Product Verification

32—Enabling Product Readiness

END PRODUCTS VALIDATION PROCESS REQUIREMENTS

33—End Products Validation

Figure 4b—Requirements for engineering a system

The developer **should**: (1) decide which of the processes in Figure 4a apply to their enterprise; (2) decide which requirements from this Standard apply for the processes selected; (3) establish appropriate policies and procedures that govern project implementation; (4) define appropriate tasks for each of the selected requirements; and (5) establish the methods and tools to support task implementation. Representative tasks, along with their expected outcomes, are provided in Annex C for each requirement of this Standard.

NOTES

1 The developer can be an enterprise, a group of enterprises, an organization, or a project.

2 A developer can be either an acquirer or a supplier of systems, subsystems, or end products.

3 A developer can act in both roles (acquirer and supplier) simultaneously on the same project, e.g., supplying an end product to another organization, while acquiring subsystems from a third organization.

For a system that contains product elements for which lower-tier development standards exist, or where standards or guides exist for safety, security, or other system aspects, these **should** be used in conjunction with this Standard—for example: (1) IEEE/EIA 12207 for a system that contains software, or for a stand-alone software product; and (2) ANSI/EIA-649 for configuration management.

4.1 Acquisition and Supply

The Acquisition and Supply Processes are used by a developer to arrive at an agreement with another party to accomplish specific work and to deliver required products, or with another party or parties to have work done and to obtain desired products. The parties can either be inside the developer's own enterprise (another project, functional organization, or project team), or can be in a different enterprise. The Acquisition and Supply Processes can be initiated as a result of a project go-ahead or approval decision, or by the receipt of an acquisition request, offer or directive. A project go-ahead can be given within an enterprise as a result of a market-needs analysis, technology breakthrough, a perceived market opportunity, a customer requirement, an internal project directive, or a similar stimulus.

NOTE—Although a project or development effort can be initiated by casual means, an agreement is nevertheless useful to ensure that all parties involved understand the purpose, goals, and expectations of the work.

An agreement can be between enterprises and between organizational elements within an enterprise, to include between projects, between projects and functional units, and between units within a project. The agreement within an enterprise can take the form of a work directive, work package, work authorization, or project memorandum of agreement. Agreements between enterprises can take the form of a formal contract for the delivery of a product, or a memorandum of agreement that establishes the working relationship between two or more enterprises on a common project.

Regardless of the form or purpose of the agreement, certain information **should** be included, for example:

a) Work to be performed;

b) Cost and schedule constraints;

c) Concept of operations;

d) Requirements to be satisfied, including known functional, performance, and interface requirements, attributes, and characteristics;

e) Product and data to be delivered;

f) Information pertaining to cost, schedule, planning, delivery information, product structure, packaging and handling instructions, or installation instructions;

g) Appropriate technical plans;

h) Applicable financial structure, management and authority provisions;

i) Exit criteria for relevant enterprise-based life cycle phases;

j) Identification of applicable engineering life cycle phases;

k) Required technical reviews.

NOTE—A developer can be developing a product without any contractual relationship to the user or customer (e.g., commercial product development). However, much of the information above must be available to the developing organization in order to proceed.

The role of the developer with respect to the two processes of Acquisition and Supply is shown in Figure 4.1.

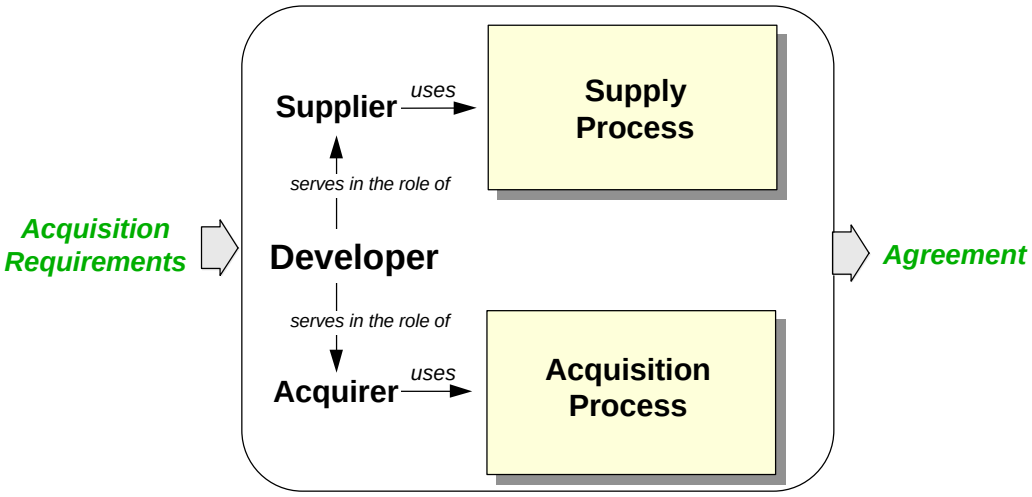

Figure 4.1—Acquisition and Supply Processes

NOTES

1 The acquirer can be either one of the following:

a) Internal to enterprise—for example, another project, marketing organization, parent project of a product team, the project team itself, executive manager, supervisor.

b) External to enterprise— for example, procurement agency, prime contractor, another developer, buyer, customer, end user, owner, purchaser.

2 The supplier can be either one of the following:

a) Internal to enterprise— for example, another project, functional organization, product team.

b) External to enterprise— for example, another developer, prime contractor, producer, seller, subcontractor, vendor.

3 The requirements of this Standard apply to the developer in its role as acquirer, supplier, or both.

4.1.1 Supply Process

This process is used by the developer when acting as a supplier to establish and satisfy an agreement with the acquirer.

Requirement 1—Product Supply

For a system, or portion thereof, supplied to an acquirer, the developer (when acting as the supplier) **shall** establish and satisfy an agreement with the acquirer.

The developer (as supplier) **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Assess the acquisition request, offer, or directive to determine the capability to meet the acquisition document requirements.

b) Establish a satisfactory agreement within legal, regulatory, enterprise, and project bounds.

c) Record the established agreement in the form appropriate to the effort.

d) Implement the processes of this Standard, as applicable, to meet the requirements of the agreement.

e) Deliver the products and other deliverables as specified in the established agreement.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement influence the Acquisition, Planning, and Control Processes.

4.1.2 Acquisition Process

This process is used by the developer when acting as an acquirer to establish an agreement with a supplier and to manage supplier performance.

The Acquisition Process includes the two requirements shown in Figure 4.1.2.

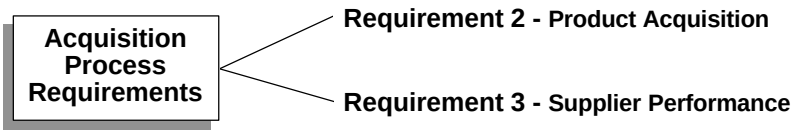

Figure 4.1.2—Acquisition Process requirements

Requirement 2—Product Acquisition

For a system, or portion thereof, acquired from a supplier, the developer (when acting as the acquirer) **shall** establish an agreement with that supplier.

The developer (as acquirer) **should** plan and do the appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Prepare the applicable acquisition request, offer, or directive to obtain supply of work or delivery of desired system products.

b) Evaluate supplier response to acquisition request, offer, or directive.

c) Make offer or provide directive to desired supplier.

d) Negotiate agreement to establish a satisfactory agreement within legal, regulatory, enterprise, and project bounds.

e) Record the established agreement in the form appropriate to the effort.

f) Accept delivered products.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement influence the Supply, Planning, and Control Processes.

Requirement 3—Supplier Performance

The developer (when acting as the acquirer) **shall** manage supplier performance to ensure that the technical effort to be accomplished by the supplier provides end products that satisfy the assigned requirements.

The developer (as acquirer) **should** plan and do the appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Define the required developer-supplier relationships.

b) Participate on appropriate supplier product teams.

c) Monitor supplier performance against key product metrics.

d) Flow-down changes in requirements or operational concept that might affect the supplier's project.

e) Control changes to requirements made by the supplier that would affect the developer's project or other related projects or products.

f) Assess supplier performance against assigned requirements, including conduct of, or participation in, appropriate technical reviews.

g) Validate products delivered from the supplier, or ensure that products have been validated before delivery and prior to integration with other products that form a composite end product intended to meet the developer's specified requirements.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement influence the Planning, Assessment, Control, and Implementation Processes.

4.2 Technical Management

The Technical Management Processes are to be used to plan, assess, and control the technical work efforts required to satisfy the established agreement. The relationship of the three Technical Management Processes for planning, assessing progress, and controlling the technical effort is shown in Figure 4.2

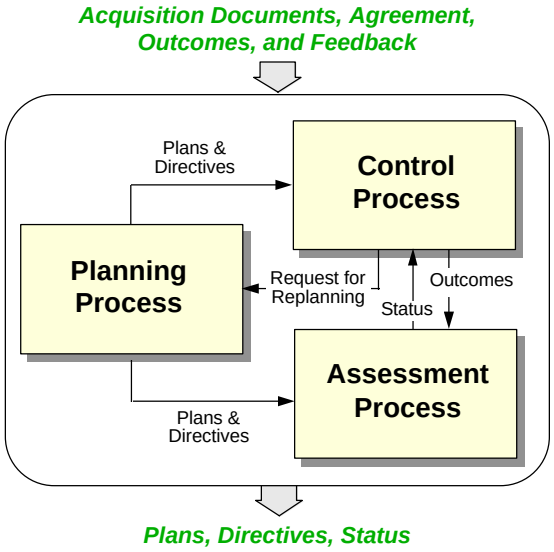

Figure 4.2—Technical Management Processes

4.2.1 Planning Process

This process is used to support enterprise and project decision making and to prepare necessary technical plans that support and complement project plans to: (1) arrive at a decision to supply services according to an external solicitation; (2) determine whether to proceed with an internal enterprise project for a new product or a product improvement; (3) guide the work efforts that will meet the requirements of an established agreement; or (4) replan applicable processes for engineering a system. Replanning is normally initiated (1) when required by an agreement; (2) when significant variations or anomalies are identified from other Technical Management process outcomes, or (3) before implementation of the next enterprise-based life cycle phase.

The five requirements associated with the Planning Process are shown in Figure 4.2.1.

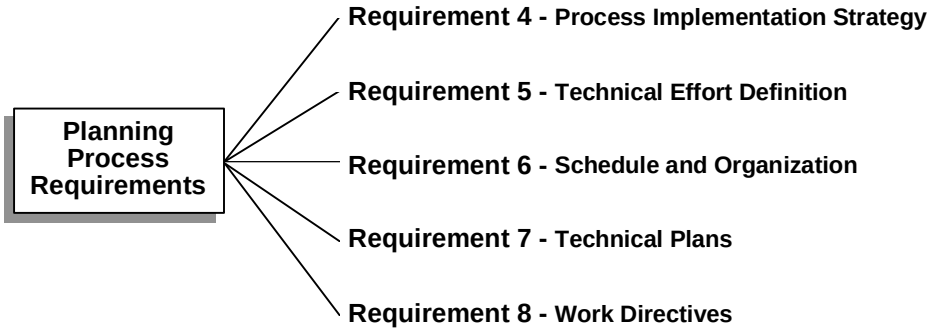

Figure 4.2.1—Planning Process requirements

> **Requirement 4—Process Implementation Strategy**
>
> The developer **shall** define a strategy for implementing the adopted processes of this Standard as a basis for project technical planning and that is in accordance with the agreement.

The developer **should** plan and do the appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identify stakeholders who will have an interest or stake in the outcome of the project.

b) Identify and acquire applicable documents, and the requirements therein, that could affect the project.

c) Identify associated process approaches for development, production, test, deployment/installation, training, support, and disposal that will require enabling products to be developed or procured.

d) Identify applicable enterprise-based life cycle phases (see Annex B), expected work product outputs, applicable management reviews, and life-cycle-phase exit criteria.

e) Identify and define how the applicable processes of this Standard will be integrated, how internal and external projects will be involved, and how they will be integrated.

f) Identify and define progress assessment metrics and reporting requirements.

g) Prepare, document, and make available the process implementation strategy.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement: (1) provide a roadmap for the technical implementation of the project, including engineering life-cycle activities within specified enterprise-based life cycle phases; (2) are to be implementable by each product team or product manager; (3) are used in preparing and negotiating an agreement; and (4) influence the developer's ability to fulfill the other requirements of the Planning Process. The process implementation strategy includes requirements for the processes to be undertaken, applicable constraints, completion criteria, and feasibility of each process, considering resources (personnel, materials, technology) and project execution environment. This strategy can be a part of the project plan or a stand-alone document.

Requirement 5—Technical Effort Definition

The developer **shall** define a technical effort that is in accordance with the process implementation strategy.

The developer **should** plan and do the appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identify project requirements to include: agreement requirements; other stakeholder requirements; and enterprise, project, and associated process constraints.

b) Establish an information database that will allow capture of project data and be able to securely retain and make information available, as required.

c) Determine the risk management strategy to identify technical risks to the appropriate level and properly avert those risks that could adversely affect the project.

d) Define product metrics by which the quality of the products will be evaluated and process metrics by which the efficiency and effectiveness of the technical effort will be measured.

e) Establish cost objectives (e.g., ownership, acquisition, operating, support, and disposal) to be used in tradeoff analyses.

f) Identify technical performance measures that will be used to determine the success of the system, or portion thereof, and that will receive management focus and be tracked using Technical Performance Measurement (TPM) procedures.

NOTE—A TPM program provides an early warning of the adequacy of a design in terms of satisfying selected critical performance parameter requirements of a system end product. TPM also examines marginal cost benefit of performance in excess of requirements. A *critical performance parameter* is one that characterizes a significant total system qualifier. In addition, it must be possible to project the evolution of the parameter as a function of time toward the desired value at the completion of development. The projection can be based on verification, validation, planning, or historical data.

g) Identify applicable tasks based on analysis of the key events of the project and entry and exit criteria for each event.

h) Identify the appropriate methods and tools, required facilities and equipment, and training required to be able to complete defined tasks and meet event exit criteria.

i) Identify applicable or potential technology constraints and develop an approach for overcoming each constraint by using an appropriate mitigation approach and by technology insertion at the appropriate time in the enterprise-based life cycle.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement will provide guidance for preparing schedules and applicable technical plans and for identifying resource requirements, and will influence the developer's ability to complete the other applicable processes for engineering a system.

Requirement 6—Schedule and Organization

The developer **shall** schedule and organize the defined technical effort.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Develop event-based schedules based on key events, related tasks, and relevant completion criteria for the applicable enterprise-based life cycle phase.

b) Develop the calendar-based schedule, showing the dates of expected task and event completion and the dependency relationships among tasks.

c) Identify resources required to complete the scheduled tasks.

d) Define the staffing and discipline needs to complete the scheduled tasks, training needs, and risks if required staff are not available.

e) Define the team and organizational structure to complete the scheduled tasks within resource constraints.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide guidance for preparing applicable technical plans used to guide completion of the technical efforts for each applicable process to meet agreement requirements.

Requirement 7—Technical Plans

The developer **shall** create technical plans to ensure an integrated and cost effective technical effort in accordance with the defined schedule and organization.

The developer **should** prepare appropriate plans to complete this requirement. Plans to consider include the following:

a) Engineering Plan;

b) Risk Management Plan;

c) Technical Review Plan;

d) Verification Plans;

e) Validation Plans;

f) Other applicable plans as called for in the agreement or by enterprise policies and procedures.

The expected outcomes for the tasks related to developing these plans are provided in Annex C. The outcomes associated with completing this requirement provide guidance for preparing work directives and completing other applicable project processes for engineering a system.

Any plan created **should** include the scope, tasks, methods, tools, metrics, risks, and resources as applicable to fulfill the purpose of the plan.

NOTE—Annex D of this Standard contains a listing of typical planning documents. Some projects require either more or significantly less documentation. These planning documents can be tailored as to the level and formality of planning to suit project complexity and uncertainty.

Requirement 8—Work Directives

The developer **shall** create work directives that implement the planned technical effort.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Develop individual project team or organization work packages that describe the work to be done, resource sources, schedules, budget, and reporting requirements.

b) Generate work authorizations for the team or organization that provide approval for applicable teams or organizations to complete their work package requirements and to release applicable resources.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide the means to implement the planned technical effort.

4.2.2 Assessment Process

The Assessment Process is used to: (1) determine progress of the technical effort against both plans and requirements; (2) review progress during technical reviews; and (3) support control of the engineering of a system. The product and process metrics selected for assessing progress **should** provide information for risk aversion, meaningful financial and non-financial performance, and support of project management.

NOTE—When variations are sufficiently significant or cannot be corrected by re-accomplishment of the process tasks that generated the outcome data, the Planning Process is re-initiated in order to implement appropriate corrective actions.

The three requirements associated with the Assessment Process are shown in Figure 4.2.2.

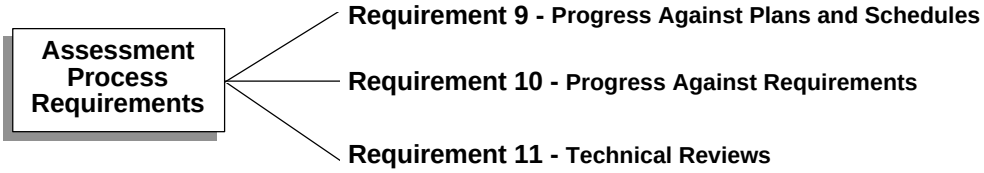

Figure 4.2.2—Assessment Process requirements

Inputs to the Assessment Process are in the form of technical plans, stakeholder requirements, and engineering outcomes from other processes.

Requirement 9—Progress Against Plans and Schedules

The developer **shall** assess the progress of the technical effort against applicable technical plans and schedules.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identify events, tasks, and process metrics for monitoring progress against plans and schedules.

b) Collect and analyze identified process metrics data and results from completion of planned and scheduled tasks and events.

c) Compare process metrics data against plans and schedule to determine technical areas requiring management or team attention.

d) Determine and implement required changes to correct variances, make changes to plan and schedule, and redirect work.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide status information to enable efficient use of resources, evaluation of progress against plan, identification of variances of cost and schedule from planned project management baselines, and early identification and resolution of productivity problems.

NOTE—*Process metrics* are identified and used to assess the means of attaining stakeholder satisfaction. Process metrics include earned value (cost/schedule measure), amount of waste, number of engineering changes, percentage of drawings completed, number of drawing errors, percentage of lines of code completed, rework percentage, idle time (e.g., work in progress), change rate, and turnover in personnel. The criteria for process metric selection are based on how well enhancement in project performance correlates with improvement in potential customer satisfaction.

Requirement 10—Progress Against Requirements

The developer **shall** assess the progress of system development by comparing currently defined system characteristics against requirements.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identify product metrics, and their expected values, that will affect the quality of the product and provide information of the progress toward satisfying acquirer and other stakeholder requirements, as well as derived requirements.

b) Collect and analyze product metrics data.

c) Record rationale for decisions and assumptions made with respect to collected data.

d) Compare results against requirements to determine degree of technical requirement satisfaction, progress toward maturity of the system (or portion thereof) being engineered, and variations and variances from requirements.

e) Identify required changes and implement approved revisions to specifications and configuration baselines.

The expected outcomes for these representative tasks are provided in Annex C. Representative outcomes associated with completing this requirement provide: (1) an evaluation of the progress toward meeting requirements pertaining to the system being engineered or reengineered; (2) status information to enable efficient use of resources, (3) evaluation and tracking of system quality and technology, (4) faster response time to inquiries from acquirer or other stakeholders, (5) identification of variances from planned improvements in critical technical parameters as the design evolves, (6) early identification and resolution of system related problems, and (7) tracking of tradeoff analysis recommendations, effectiveness analysis results, verification outcomes, and validation results.

NOTE—*Product metrics* are used to measure stakeholder satisfaction, deliver an ever-improving value to the acquirers of system end products, and be indicative that the design process is continuing toward an acceptable solution. An example of an input product metric is the quality of materials and skills of assigned project personnel. An example of an output metric is a technical performance measure (TPM).

Requirement 11—Technical Reviews

The developer **shall** conduct technical reviews of progress and accomplishments in accordance with appropriate technical plans.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identify the review objectives and requirements cited in the Technical Review Plan, enterprise policies and procedures, and agreement, as applicable.

b) Determine progress toward satisfying the technical review entry requirements.

c) Establish the technical review board, agenda, and speakers.

d) Prepare the appropriate materials to include the read-ahead technical review package and presentation package.

e) Facilitate and support identification and resolution of emerging issues prior to the review.

f) Conduct the technical review according to the Technical Review Plan, identifying and documenting action items required to meet the review objectives.

g) Close out the review after (1) minutes have been prepared, approved, and distributed, (2) action items have been resolved, and (3) the review has been signed off by the chairperson.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completion of this requirement (1) help ensure that all event-based plan criteria have been met, (2) provide ongoing status of design maturity and how well the concepts satisfy requirements, (3) provide traceability of requirements and validity of assumptions and decision rationale, (4) provide identification of issues to be resolved and those issues not determined during the development effort, and (5) highlight related risks, needed resources, and preparation for conducting the next engineering life cycle phase development effort.

Representative technical reviews, used for assessing progress against the requirements and technical plans and for assessing planned tasks, are described in Annex E.

4.2.3 Control Process

The Control Process is used to: (1) manage the conduct and outcomes of the Acquisition and Supply Processes, System Design Processes, Planning and Assessment Processes, Product Realization Processes, and Technical Evaluation Processes; (2) monitor variations from the plan and anomalies relative to requirements; (3) distribute required and requested information; and (4) ensure necessary communications. This process supports satisfaction of the agreement and assurance that variations and anomalies are corrected by repeating appropriate tasks.

The two requirements associated with the Control Process are shown in Figure 4.2.3.

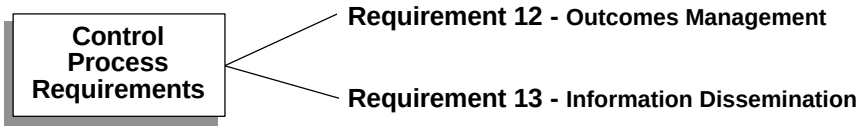

Figure 4.2.3—Control Process requirements

Inputs to the Control Process are in the form of outcomes from other processes plus project and enterprise information affecting the engineering of a system.

Requirement 12—Outcomes Management

The developer **shall** manage the outcomes of the technical effort.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Capture the outcomes, descriptions of methods and tools used, decisions and assumptions, lessons learned, and other data that allow for tracking requirements.

b) Perform configuration management in accordance with the Configuration Management Plan.

c) Perform change management in accordance with the Change Management Plan.

d) Perform interface management in accordance with the Interface Management Plan.

e) Perform risk management in accordance with the Risk Management Plan.

f) Perform data and document management in accordance with the Data and Document Management Plan.

g) Manage the information database to ensure that captured data is properly retained, is secure, and is available to those with authority to have access.

h) Manage and track stakeholder requirements, system technical requirements, logical solution representations, physical solution representations, derived technical requirements, specified requirements, approved changes, and validation results.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement help to ensure that the outcomes of the applicable processes for engineering a system are properly recorded and managed according to the applicable plan, the agreement, or enterprise policies and procedures.

Requirement 13—Information Dissemination

The developer **shall** ensure that required and requested information is disseminated in accordance with the agreement, project plans, enterprise policies, and enterprise procedures.

The developer **should** plan and do appropriate tasks to complete this requirement. Information to consider for dissemination includes, as appropriate, the following materials captured and controlled in the information database:

a) Progress status;

b) Planning information;

c) Requirements;

d) Technical data package and other materials for technical reviews;

e) Design data and schema;

f) Lessons learned;

g) Variances and anomalies from validations and verifications and other progress assessments;

h) Data deliverables;

i) Approved changes;

j) Work directives resulting from management decisions, planning, or approved changes.

The expected outcomes for the representative tasks associated with this distribution are provided in Annex C. The outcomes associated with completing this requirement help to ensure that required and requested information is appropriately distributed to satisfy the needs of the acquirer and requesters, and in accordance with an agreement, project directives and plans, and enterprise policies and procedures.

4.3 System Design

The System Design Processes are used to convert agreed-upon requirements of the acquirer into a set of realizable products that satisfy acquirer and other stakeholder requirements.

Two processes are involved—Requirements Definition and Solution Definition. The relationship of these processes is shown in Figure 4.3.

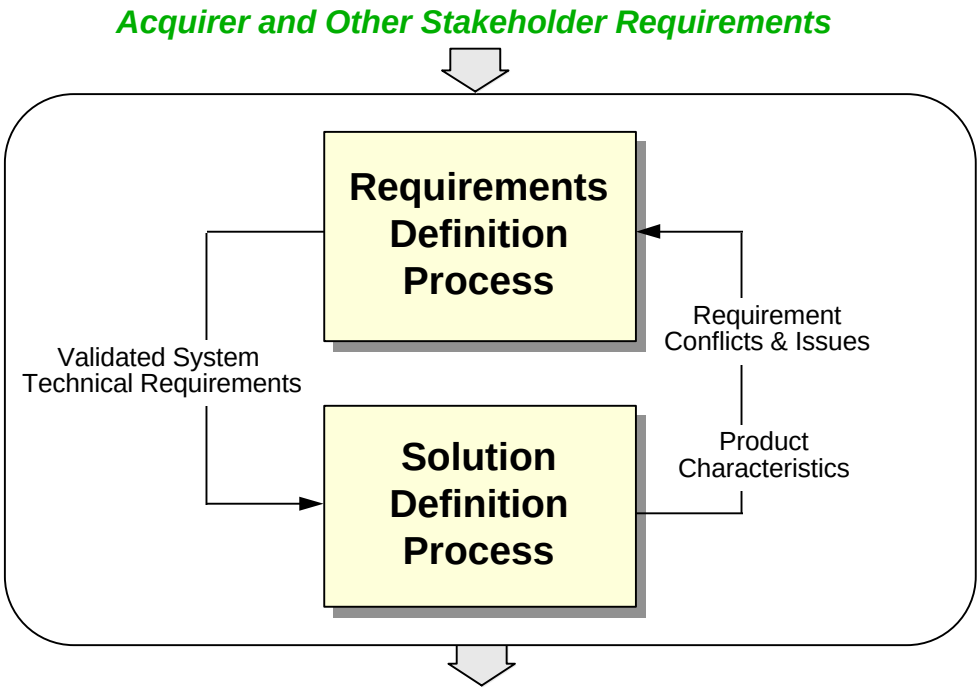

Figure 4.3—System Design Processes

This technical effort begins with the identifying, collecting, and defining *acquirer* and *other stakeholder requirements*. These requirements are transformed into a set of validated *system technical requirements*. The validated system technical requirements are then transformed into a design solution described by a set of *specified requirements*. The specified requirements take the form of specifications, drawings, models, or other design documents, depending on design maturity. These are used to: 1) build, code, assemble and integrate end products; 2) verify end products against; 3) obtain off-the-shelf products; or 4) assign to a supplier for development of subsystem products. The relationship between the requirements involved with the System Design Processes is shown in Annex G.

NOTE—Requirements traceability is instituted for tracking requirements from the identification of acquirer and other stakeholder requirements to the system technical requirements, logical solution representations, physical solution representations, derived technical requirements, and specified requirements. (See Requirement 12, task h.)

4.3.1 Requirements Definition Process

The three requirements associated with the Requirements Definition Process are shown in Figure 4.3.1.

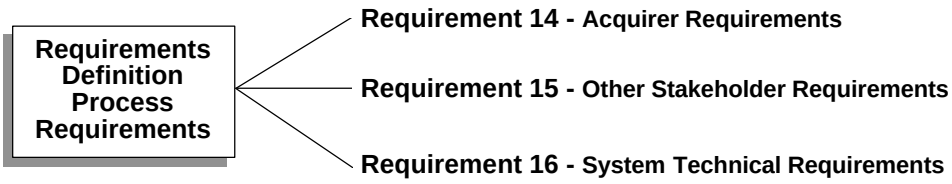

Figure 4.3.1—Requirements Definition Process requirements

Inputs to the Requirements Definition Process are of three types: (1) requirements from the agreement, other documents, and individuals or groups that have a stake in the outcome of the engineering or reengineering of the system, (2) requirements in the form of outcomes from other processes such as technical plans and decisions from technical reviews, and (3) requested or approved changes to requirements of the first type.

NOTES

1 The requirements defined by this process come from stakeholders who have an interest in the system being engineered. Stakeholders are of two kinds: the acquirer of the system products (see the definition of acquirer in the Glossary, Annex A) and all other stakeholders (see the definition of other stakeholder in Annex A).

2 The Requirements Definition Process is used to transform stakeholder requirements into a set of system technical requirements. These requirements are stated in acceptable technical terms and represent a reasonably complete description of the problem that must be solved to provide a set of end products and enabling products that meet the acquirer's and other stakeholders' needs and expectations.

3 The Requirements Definition Process is re-accomplished, as necessary, whenever requirements in an agreement change or when other stakeholder requirements are identified that affect the product design or otherwise constrain the technical effort required to engineer a new system, develop a derivative system, or reengineer a legacy system. Such changes could be caused by technology limitations, project schedule and cost anomalies, or new requirements.

4 Sometimes it is important to preserve competition when defining requirements to ensure that there will be more than one supplier that can meet the requirements. Otherwise, the cost of a single supplier can be too high since there can sometimes be little incentive to give a low-cost bid.

Requirement 14—Acquirer Requirements

The developer **shall** define a validated set of acquirer requirements for the system, or portion thereof.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identify, collect, and prioritize assigned, customer, user, or operator requirements for the system, or portion thereof, including any requirements for development, production, test, deployment/installation, training, operations, support/maintenance, and disposal of the system's products.

b) Ensure that the resulting set of requirements agrees with the acquirer needs and expectations (see Requirement 26).

c) Record the resulting set of acquirer requirements in the established information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement are used, when combined with other stakeholder requirements, to define the system technical requirements and to identify requirements for enabling products.

Requirement 15—Other Stakeholder Requirements

The developer **shall** define a validated set of other stakeholder requirements for the system, or portion thereof.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identify and collect other stakeholder requirements that can constrain the system's end products.

4 Constraints can result, for example, from treaties, laws, regulations, standards, culture, natural laws, or firm customer or user needs.

5 Constraints also apply to those characteristics necessary to interface with other existing systems.

Requirement 16—System Technical Requirements

The developer **shall** define a validated set of system technical requirements.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Establish required transformation rules, priorities, inputs, outputs, states, modes, and configurations, as appropriate to each system product.

b) Define operational requirements to include operational profiles, and for each operational profile the utilization environment, events to which system end products must respond, frequency of use, physical and functional interfaces, and system functional requirements (what system end products must accomplish).

c) Define the performance requirement (how well each functional requirement must be accomplished), including identification of critical performance parameters.

d) Analyze acquirer and other stakeholder requirements to define human factor effects and concerns, establish capacities and timing, define technology and product design constraints, define enabling product requirements, identify conflicts, and determine criteria for tradeoff analyses to resolve conflicts.

e) Identify and resolve requirements that have questionable utility or have unacceptable risk of not being satisfied.

f) Resolve identified conflicts between the sets of acquirer requirements and other stakeholder requirements, and among these sets (see Requirement 23).

g) Prepare a set of system technical requirement statements that are well formulated in accordance with Requirement 25.

h) Ensure that the set of system technical requirements is correct in accordance with Requirement 28.

i) Record the resulting set of system technical requirements in the established information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide a set of system technical requirements that are unambiguous, complete, consistent, achievable, verifiable, and necessary and sufficient for a system design.

4.3.2 Solution Definition Process

The Solution Definition Process is used to generate an acceptable design solution. This solution satisfies: 1) the system technical requirements resulting from completing the Requirements Definition Process described in Subclause 4.3.1; and 2) the derived technical requirements from the Solution Definition Process described in this subclause.

The three requirements associated with the Solution Definition Process are shown Figure 4.3.2.

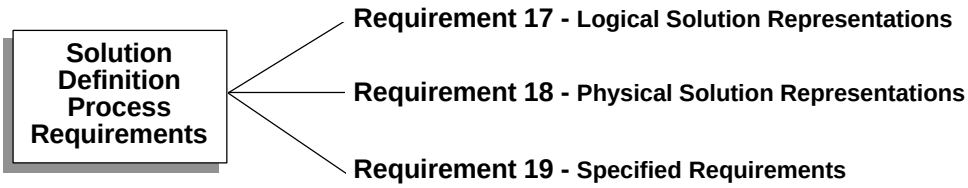

Figure 4.3.2—Solution Definition Process requirements

NOTE—The purpose of the requirements related to the Solution Definition Process is to solve the technical problem. This involves identifying alternative end products for the system, selecting and defining an optimal set of end products, defining the feasible subsystems related to the end products, identifying requirements for enabling products, and identifying needed high-risk technology developments.

Requirement 17—Logical Solution Representations

The developer **shall** define one or more validated sets of logical solution representations that conform with the technical requirements of the system.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Select and implement one or more appropriate approaches to providing an abstract definition of the solution to the system technical requirements. For the approaches selected, complete the appropriate tasks from (b) through (d) below that aid in defining logical solution representations.

NOTE—Functional analysis, object-oriented analysis, structured analysis, and information engineering analysis are recognized approaches found in text books and other literature to develop logical solution representations in terms of, for example, functional flows, behavioral responses, state and mode transitions, timelines, control flows, data flows, information models, object services and attributes, context diagrams, threads, data structures, and functional failure modes and effects.

b) Establish sets of logical solution representations by (1) doing tradeoff analyses (see Requirement 23), (2) identifying and defining interfaces, states and modes, timelines, and data and control flows, (3) analyzing behaviors, and (4) analyzing failure modes and defining failure effects.

c) Assign system technical requirements (especially performance requirements and constraints from the system technical requirements) to elements of the logical solution representations, e.g., subfunctions, groups of subfunctions, objects, and data structures.

NOTES

1 There can also be *system technical requirements* that are neither appropriate to assign to the sets of *logical solution representations* nor modifiable into *derived technical requirements*. An example is a characteristic or constraint applicable only to the system, not to the products of the system. These *system technical requirement* statements must be analyzed and assigned during Requirement 18 tasks a), b), and c).

2 There will be additional derived technical requirements prepared to reflect systems analysis results from Requirement 18 task c).

d) Identify and define derived technical requirement statements resulting from tasks a) and b). Ensure that the derived technical requirements are stated acceptably in accordance with Requirement 25.

e) Ensure that each set of logical solution representations is correct in accordance with Requirement 29.

f) Record the resulting sets of logical solution representations, the set of derived technical requirement statements, and any unassigned system technical requirements [see the note under task c) above], along with source rationale and assumptions in the established information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement, when combined with the system technical requirements, provide the basis for developing alternative physical solution representations.

NOTES

1 Conditions for logical groupings are determined by many factors, and vary from one project to another. One common driver for logical groupings is to enable the use of existing products, and thus lessen development time and cost. Another common reason is to gain some advantage by introducing a particular new technology. In either of these cases, the grouping can result in interfaces that did not previously exist. New requirements have to be derived to accommodate these.

2 Accomplishment of the tasks associated with this requirement is often iterative because outcomes raise questions that require certain tasks of the Requirements Definition Process to be reaccomplished. In turn, certain tasks associated with defining logical solution representations and derived technical requirements are reaccomplished. Such iteration is important in order to lessen the possibility of more costly iterations of System Design Processes during a later engineering life cycle phase.

Derived technical requirements and requirements associated with logical solution representations **should** be incorporated into traceability procedures. This will enable ensuring that system technical requirements are properly supported by the derived technical requirements and logical solution representations.

Requirement 18—Physical Solution Representations

The developer **shall** define a preferred set of physical solution representations that agrees with the assigned logical solution representations, derived technical requirements, and system technical requirements.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Analyze logical solution representations, derived technical requirements, and any unassigned system technical requirements [see the note under Requirement 17, task c)] to determine which ones (1) provide requirements for enabling products, (2) can be done best manually or by facilities, materials, data, services, or techniques, and (3) can be done best by hardware, software, or firmware products (new or existing).

b) Assign representations from Requirement 17, unassigned system technical requirements, and derived technical requirements to physical entities that will make up a physical solution.

> NOTE—The assignment to physical entities and the generation of alternative solutions composed of these entities are tightly coupled and iterative.

c) Generate alternative physical solutions by:

   1) Identification and definition of physical interfaces
   2) Identification and analysis of critical parameters
   3) Identification and assessment of physical solution options:
      (a) Technology requirements
      (b) Off-the-shelf availability
      (c) Competitive considerations
      (d) Failure modes, effects, and criticality
      (e) Performance assessment
      (f) Life cycle considerations
      (g) Capacity to evolve
      (h) Make versus buy
      (i) Standardization considerations
      (j) Integration concerns
   4) Performance of systems analysis (see Requirements 22, 23, and 24)

d) Identify and define derived technical requirement statements resulting from tasks a), b), and c) that are stated acceptably in accordance with Requirement 25.

e) Select the preferred physical solution representation for further characterization into a design solution from the evaluation of each physical solution representation results (Requirements 22, 23, and 24.

f) Ensure that the selected physical solution representation is consistent with the assigned logical solution representations, derived technical requirements, and any unassigned system technical requirements [see the note under Requirement 17, Task c)].

g) Record the selected physical solution representation, and the outcomes of Task d) above, along with selection rationale and assumptions, in the established information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide the preferred physical solution representation that will be fully characterized during Requirement 19. Additionally, the outcomes show that

1) the preferred physical solution representation satisfies the assigned requirements of the logical solution representations, derived technical requirements, and system technical requirements.
2) the preferred physical solution representation is upward- and downward-traceable with respect to the assigned requirements of logical solution representations, derived technical requirements, and any unassigned system technical requirements [see the note under Requirement 17, task c)].

Outcomes can be displayed as a hierarchical structure of physical entities, schematics, physical models, analytical models, or explosion diagrams.

NOTES

1 As each physical solution representation is defined, it usually is necessary to reaccomplish tasks related to the definition of logical solution representations to ensure that the final set of derived requirements and requirements associated with logical solution representations is traceable to the preferred physical solution representation, and vice versa.

2 Physical solution representations will eventually be composed of one or more of the following: hardware, software, firmware, material, data (e.g., manuals, handbooks), facilities, people, services, techniques, processes, and manual procedures.

Requirement 19—Specified Requirements

The developer **shall** specify requirements for the design solution.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Fully characterize the design solution.

b) Ensure that the design solution is consistent with its source requirements (selected physical solution representation requirements, associated system technical requirements, and derived technical requirements) in accordance with Requirement 30

c) Specify requirements (including functional and performance requirements, physical characteristics, and test requirements) for the system, system end products, and subsystems of each end product, as applicable to the engineering life cycle phase.

d) Record the design solution work products, including the specified requirements, in the established information database with all tradeoff analyses results, design rationale, assumptions, and key decisions to provide traceability of requirements up and down the system structure.

e) Establish projects to engineer enabling products that require development, or to procure those that are off-the-shelf or will be reused, that will satisfy identified requirements for associated processes (production, test, deployment/installation, training, support or maintenance, and retirement or disposal) related to the system's end products.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide a fully characterized design solution that (1) can be implemented

through further development of subsystems, off-the-shelf procurement or reuse, coding, or fabrication, and (2) provide the basis for the assembly and integration of subsystem products into end products required for verification.

NOTE— A fully characterized design solution can be in terms of, as appropriate: (1) specifications for the system, end products, subsystems, and applicable interfaces; (2) interface control drawings or descriptions, detailed drawings, or sketches; and (3) parts lists, data dictionaries, or other planned physical configuration records.

4.4 Product Realization

The Product Realization Processes are used to: (1) convert the specified requirements and other design solution characterizations into either a verified end product or a set of end products in accordance with the agreement and other stakeholder requirements; (2) deliver these to designated operating, customer, or storage sites; (3) install these at designated operating sites or into designated platforms; and (4) provide in-service support, as called for in an agreement.

The two processes related to Product Realization are shown in Figure 4.4.

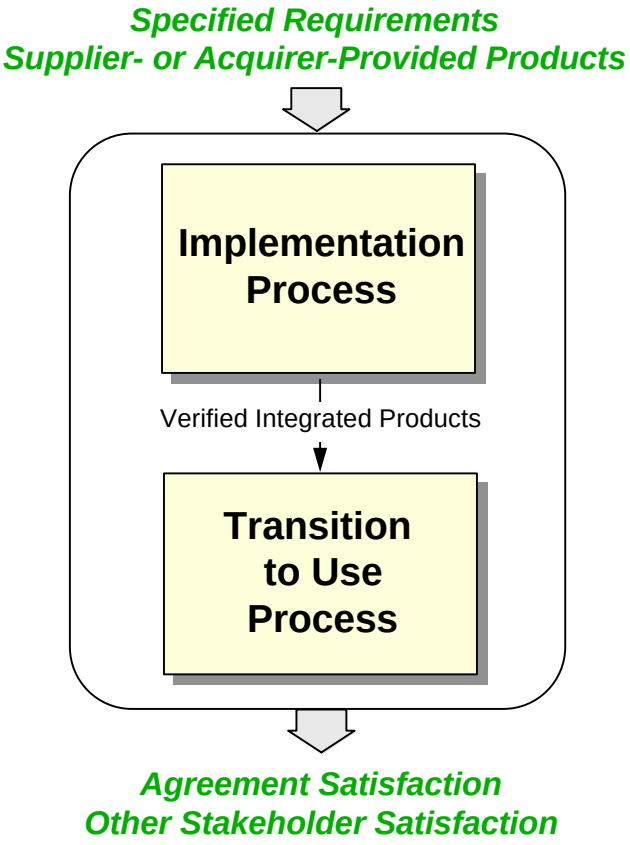

Figure 4.4—Product Realization Processes

4.4.1 Implementation Process

One requirement is associated with the Implementation Process. It requires transforming the characterized design solution into an integrated end product that conforms to its specified requirements.

Requirement 20—Implementation

The developer **shall** implement the design solution in accordance with the specified requirements to obtain a verified end product.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Receive from suppliers, reuse from off-the-shelf supply, or receive from the acquirer (e.g., customer-furnished items) the subsystem products that make up the system's end products, or, as appropriate, code or build the end products according to specified requirements and detailed drawings or other design documentation.

NOTE— Requirement 3, Supplier Performance, is invoked whenever subsystem products are acquired from suppliers or lower-tier developers outside the enterprise, as well as when the supplier is an organizational entity within the developer's own enterprise.

b) Validate the subsystem products received or reused against their acquirer requirements (input requirements to the subsystem product development) using the End Products Validation Process, Requirement 33, unless (1) the supplier validated the products prior to delivery, as required in the agreement, or (2) the reused products have already been validated. Proof of validation is needed for both conditions.

c) Assemble the validated subsystem products, or physically integrate such products, into the respective test article or end product to be verified.

d) Verify each test article or end product against its specified requirements (output requirements of the system end product development) in accordance with Requirement 31.

e) Ensure, in accordance with Requirement 32, that the enabling products for each associated process will be ready and available to perform their intended support functions required by the system's end products.

NOTE—The relevant end products for enabling products are verified and validated as necessary during development of the building block related to the enabling product (see Clause 6).

f) Validate the verified end products against their acquirer requirements (input requirements to system end product development) prior to delivery, if required by the agreement, in accordance with Requirement 33.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide a fully integrated end product that (1) satisfies its specified requirements, and (2) if required to be validated prior to delivery, conforms to its related acquirer requirements.

End product physical integration **should** ensure that: (1) internal and external interfaces for the composite end product (including user, operator and maintainer interfaces, and voice/data communications) function according to specified requirements; (2) defined states, modes, dynamic allocations or other operational switching functions perform as required; and (3) any designed overload conditions, reduced operational levels, or designed-in degraded mode of operations are included.

4.4.2 Transition to Use Process

The Transition to Use Process results in products delivered to the appropriate destinations, in the required condition for use by the acquirer, and for the appropriate training of installers, operators, or maintainers of the products.

Requirement 21—Transition to Use

The developer **shall** transition verified products to the acquirer of the products in accordance with the agreement.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Acquire and put in place appropriate enabling products to carry out relevant transition to use requirements.

b) Prepare, as required by the agreement, end products for shipping and storage.

c) Store end products awaiting shipping and, in accordance with the agreement, ship or transport to the acquirer at the intended usage sites.

d) Prepare, as required by the agreement, sites where end products will be stored, installed, used or maintained, or serviced.

e) Install end products, as required by the agreement, at the appropriate sites.

f) Perform commissioning, as required by the agreement, to bring delivered or installed end products to operational readiness with appropriate acceptance and certification tests completed in accordance with Requirement 33.

g) Provide, if required by the agreement, a parallel operation (ghosting) of the new and the legacy end products so that service is continuous during the transition period.

h) Provide, in accordance with the agreement, training for users, maintenance, and other personnel.

i) Provide, in accordance with the agreement, in-service support.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement fulfill the delivery requirements of the agreement.

NOTE—Transition to Use tasks will be dependent on whether the end product is being delivered for intended marketplace use or sale, or if the end product is delivered to another developer for integration into a set of other end products to make up an end product higher in the system structure.

4.5 Technical Evaluation

The Technical Evaluation Processes are intended to be invoked by one of the other processes for engineering a system. Four processes are involved: Systems Analysis, Requirements Validation, System

Verification, and End Products Validation. The relationship between these processes is shown in Figure 4.5.

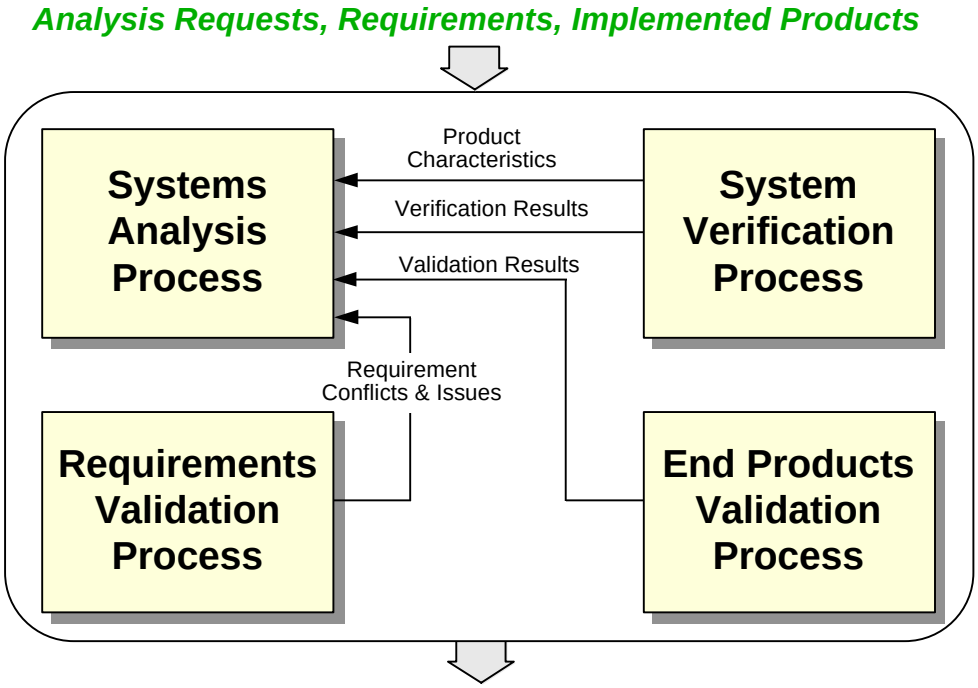

Figure 4.5—Technical Evaluation Processes

4.5.1 Systems Analysis Process

The Systems Analysis Process is used to: (1) provide a rigorous basis for technical decision making, resolution of requirement conflicts, and assessment of alternative physical solutions; (2) determine progress in satisfying system technical and derived technical requirements; (3) support risk management; and (4) ensure that decisions are made only after evaluating the cost, schedule, performance, and risk effects on the engineering or reengineering of the system.

The three requirements associated with the Systems Analysis Process, when invoked by other processes in this Standard, are shown in Figure 4.5.1.

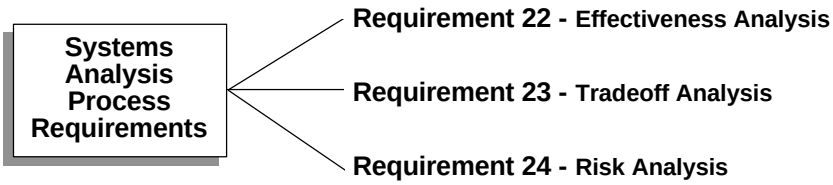

Figure 4.5.1—Systems Analysis Process requirements

Completion of Systems Analysis requirements **should** ensure, as appropriate, that

a) the effectiveness of each design solution is appropriately evaluated;

b) the effect on any interfacing products or platforms is evaluated for each alternative solution, in time to support selection among these alternatives (this includes interoperability and integration effects of electronic interference and communication, as well as functional, human, and physical interfaces);

c) cost (e.g., unit production cost, developmental cost, and/or life-cycle cost) is appropriately treated as an assigned requirement or as an independent variable when conducting tradeoffs with technical requirements;

d) cost or price, schedule, performance, and risk effects of each functional, performance, and design alternative are defined, calculated, and reported;

e) estimated total ownership costs including hidden cost effects (for example, from manufacturing processes variability; excessive precision of manufacturing or test processes; special materials, finishes, and painting of products; and product complexity), the costs of operation, and all associated processes are known;

f) primary functional characteristics of solutions (for example, producibility, testability, deployability, operability, supportability, trainability, and disposability) are directly traceable to the functional and performance requirements they were designed to fulfill;

g) applicable product dependability factors such as availability, maintainability, reliability, safety, and security are not degraded;

h) projected environmental impacts are known;

i) design assumptions are valid and reasonable;

j) technology limits are recognized and understood;

k) requirements can be validated and specifications verified in a cost-effective manner.

The developer **should** identify, acquire, or develop, and implement models, including prototypes and simulations as applicable, to accomplish effectiveness analyses, do tradeoff analyses, and complete risk analyses as invoked by processes in this Standard.

Requirement 22—Effectiveness Analysis

The developer **shall** perform effectiveness analyses to provide a quantitative basis for decision making.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Plan effectiveness analyses to include purpose, objectives, execution and data collection, schedule of tasks, resource needs and availability, and expected outcomes.

b) Analyze each alternative for system and cost effectiveness, based on factors such as accuracy, availability, capacity, maintainability, reliability, responsiveness, operability, safety, security, spares requirements, survivability, transportability, and vulnerability.

c) Analyze each alternative for total ownership cost to the enterprise and to the acquirer.

d) Analyze the environmental impact of each alternative, including applicable environmental statutes and hazardous material lists, from an enterprise-based life cycle perspective (see Annex B).

e) Analyze each alternative, for each required operational profile, to provide an analytical confirmation that the alternative satisfies appropriate requirements.

f) Record effective analysis outcomes in the established information database, including assumptions, details of the analysis, findings, lessons learned, models used, rationale for decisions made, and other pertinent information that affects the interpretation of the effectiveness analysis results.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement are used, as appropriate, to: (1) assess each alternative physical solution representation, (2) assist in choosing the preferred physical solution representation, and (3) provide assessments for tradeoff analyses to aid in determining recommended decisions and their effects.

Requirement 23—Tradeoff Analysis

The developer **shall** perform tradeoff analyses to provide decision makers with recommendations, predictions of the results of alternative decisions, and other appropriate information to allow selection of the best course of action.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Plan tradeoff analyses to include:

1) the availability of required resources, execution and data collection requirements, expected outcomes, defined conditions (triggers and rigor), level of importance, objective, schedule of tasks, and type (formal, informal or judgmental; see Table C.23, Annex C);

2) selection criteria that will determine desirability or undesirability of an alternative [for example, cost, schedule performance and risk; life-cycle outcomes; -ility concerns (e.g., producibility, testability, maintainability, supportability, disposability), size weight and power consumption; effectiveness analysis outcomes];

3) weighting factors for each selection criterion in order to distinguish its degree of importance;

4) models (representative or simulation) to be used in the tradeoff analysis;

5) options to be analyzed.

b) Perform the tradeoff analysis according to plan, and:

1) Do appropriate effectiveness analysis tasks (Requirement 22) to provide a quantitative basis for evaluating options.

2) Do appropriate risk analysis tasks (Requirement 24) to quantitatively assess the risk associated with each option.

3) Collect data and analyze it to determine the cost, schedule, performance, and risk effect of each option on the system, if implemented.

4) Evaluate options against selection criteria and weighting factors, and identify and define recommendations.

5) Communicate recommendations and impacts to appropriate decision makers.

c) Record outcomes of the tradeoff analysis in the information database, including assumptions, details of the analysis, lessons learned, models used, rationale for decisions made, recommendations and effects, and other pertinent information affecting the interpretation of the decisions made.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement are used, as appropriate, to resolve requirement conflicts during requirements definition; to assess groupings of functions, objects, etc., during definition of logical solution representations; to assess design options and alternative physical solution representations during definition of physical solution representations; to determine progress in satisfying technical requirements; and to evaluate outcomes of verifications and validations.

Requirement 24—Risk Analysis

The developer **shall** perform risk analyses to develop risk management strategies, support management of risks, and support decision making.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Identification of technical risks, and resulting project risks, based on exposure to the probability of an undesirable consequence and the effect of that consequence for each tradeoff analysis option or each physical solution representation.

b) Characterize risks by causes, possible effects or consequences, likelihood of occurrence, options for dealing with risks, how long an option is available, and coupling with other risks.

c) Prioritize risks that would likely cause harm, have the greatest effect on the system, and would require attention in the near term.

d) Evaluate ways to avert risk, and determine the cost, schedule, and performance effects on the project.

e) Define and implement a plan or approach for averting each significant risk.

f) Record the risk analysis outcomes in the information database and communicate or use risk findings and impacts, as appropriate

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement help in problem prevention, to identify the degree of risk associated with recommended decision alternatives and to support the risk management program.

4.5.2 Requirements Validation Process

Requirements Validation is critical to successful system product development and implementation. Requirements are validated when it is certain that the subject set of requirements describes the input requirements and objectives such that the resulting system products can satisfy the requirements and objectives. The Requirements Validation Process helps ensure that the requirements are necessary and sufficient for creating design solutions appropriate to meeting the exit criteria of the applicable engineering life cycle phase and of the enterprise-based life cycle phase in which the engineering or reengineering efforts occur.

The five requirements associated with the Requirements Validation Process are shown in Figure 4.5.2.

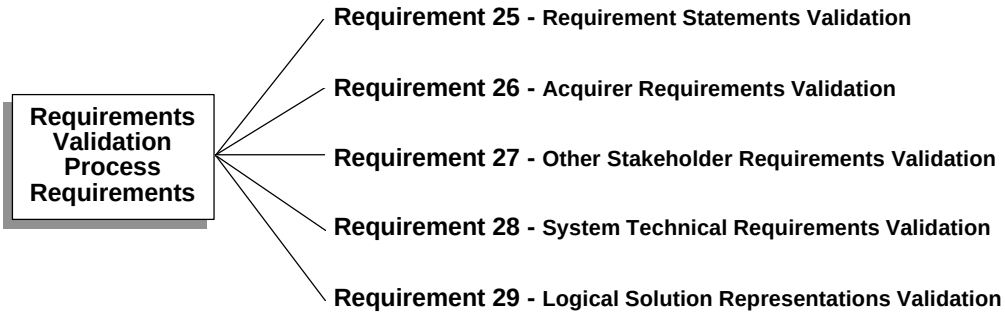

Figure 4.5.2—Requirements Validation Process requirements

One or more of these five requirements are invoked by a recommended task within either the Requirements Definition Process or the Solution Definition Process.

Requirements **should** be validated at each level of the system structure for requirements definition. Generally, validation of requirements at higher levels are a basis for validation at lower levels (see Clause 6).

Requirement 25—Requirement Statements Validation

The developer **shall** ensure that technical requirement statements and specified requirement statements, individually and as sets, are well formulated.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Analyze each requirement statement from Requirements 16, 17, 18, and 19 to ensure 1) ability to preserve competitiveness, 2) clarity, 3) correctness, 4) feasibility, 5) focus, 6) implementability, 7) modifiability, 8) removal of ambiguity, 9) singularity, 10) testability, and 11) verifiability.

b) Analyze requirement statements from Requirements 16, 17, 18, and 19 in pairs and sets to ensure 1) absence of redundancy, 2) connectivity, and 3) removal of conflicts.

The expected outcomes for these representative tasks are provided in Annex C. The validated technical requirement statements resulting from satisfying this requirement are used to guide development of system design solutions and evolve into related specified requirements.

Requirement 26—Acquirer Requirements Validation

The developer **shall** ensure that the set of defined acquirer requirements agrees with acquirer needs and expectations.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Select the methods and define the procedures for validating that the set of acquirer requirements from Requirement 14 is consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.

b) Analyze and compare identified and collected acquirer requirements to the set of defined acquirer requirements to determine downward traceability.

c) Analyze and compare the set of defined acquirer requirements to the identified and collected acquirer requirements to determine upward traceability.

d) Identify and resolve variances, voids, and conflicts.

e) Record validation results in the information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement, when combined with other stakeholder requirements, provide inputs to the definition of system technical requirements (see Annex G).

Requirement 27—Other Stakeholder Requirements Validation

The developer **shall** ensure that the set of defined other stakeholder requirements agrees with other stakeholder needs and expectations with respect to the system.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Select the methods and define the procedures for validating that the set of other stakeholder requirements from Requirement 15 is consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.

b) Analyze and compare identified and collected other stakeholder requirements with the set of defined other stakeholder requirements to determine downward traceability.

c) Analyze and compare the set of defined other stakeholder requirements with the identified and collected other stakeholder requirements to determine upward traceability.

d) Identify and resolve variances, voids, and conflicts.

e) Record validation results in the information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement, when combined with acquirer requirements, provide inputs for defining the system technical requirements (see Annex G).

Requirement 28—System Technical Requirements Validation

The developer **shall** ensure that the set of defined system technical requirements agrees with the validated acquirer and other stakeholder requirements.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Select the methods and define the procedures for validating that the set of system technical requirements from Requirement 16 is consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.

b) Analyze and compare the set of validated acquirer and other stakeholder requirements with the set of defined system technical requirements to determine downward traceability.

c) Analyze and compare the set of defined system technical requirements with the validated set of acquirer and other stakeholder requirements to determine upward traceability.

d) Analyze assumptions made with respect to defining system technical requirements to ensure that they are consistent with the system being engineered.

e) Analyze system technical requirements that have been defined as essential for the design effort or for other life-cycle considerations for which there is no parent requirement in the set of acquirer and other stakeholder requirements to ensure that they are consistent with the system being engineered and other system technical requirements.

f) Identify and resolve variances, voids, and conflicts.

g) Revalidate the system technical requirements whenever a requirement change is made that affects the acquirer requirements, other stakeholder requirements, or system technical requirements.

h) Record validation results, including lessons learned, in the information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement show that the set of system technical requirements has traceability from the set of validated stakeholders' requirements and that it is both necessary and sufficient as inputs for the definition of logical solution representations (see Annex G).

Requirement 29—Logical Solution Representations Validation

The developer **shall** ensure that each set of logical solution representations agrees with the appropriately assigned subset of system technical requirements.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Select the methods and define the procedures for validating that the sets of logical solution representations and derived technical requirements from Requirement 17 are consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.

b) Analyze and compare the set of validated system technical requirements with the sets of defined logical solution representations and derived technical requirements to determine downward traceability.

c) Analyze and compare the sets of defined logical solution representations, derived technical requirements, and any unassigned system technical requirements [see the note under Requirement 17, task c)] with the validated set of validated system technical requirements to determine upward traceability.

d) Analyze assumptions made with respect to defining sets of logical solution representations and derived technical requirements to ensure that they are consistent with the system technical requirements and the system being engineered.

e) Identify and resolve variances, voids, and conflicts.

f) Revalidate the sets of logical solution representations whenever a requirement change is made that affects the acquirer requirements, other stakeholder requirements, system technical requirements, or sets of defined logical solution representations and derived technical requirements.

g) Record validation results, including lessons learned, in the information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide derived technical requirements and logical solution representations as inputs into the definition of physical solution representations (see Annex G).

4.5.3 System Verification Process

The System Verification Process is used to ascertain that: (1) the system design solution generated by implementing Requirement 19 is consistent with its source requirements (selected preferred physical

solution representation); (2) end products at each level of the system structure implementation, from the bottom up, (see Clause 6) meet their specified requirements; (3) enabling product development or procurement for each associated process is properly progressing; and (4) required enabling products will be ready and available when needed to perform.

NOTE—Verification consists of inspections, reviews, analyses, demonstrations, tests, or service experience applied in accordance with the verification plan.

The three requirements associated with the System Verification Process are shown in Figure 4.5.3.

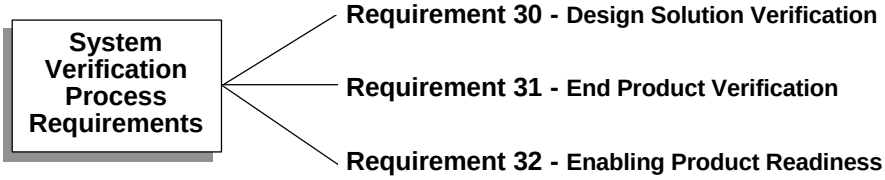

Figure 4.5.3—System Verification Process requirements

Requirement 30—Design Solution Verification

The developer **shall** verify that each end product defined by the system design solution conforms to the requirements of the selected physical solution representation.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Plan the design solution verification in accordance with the Verification Plan, agreement, applicable enterprise-based life cycle phase, and level in the system structure. Include

   1) selection and definition of the appropriate method for design solution verification;

   2) verification procedures to be followed for the method selected; the purpose and objective of each procedure, pretest action, and post-test action; and the criteria for determining the success or failure of the procedure;

   3) establishment and checkout (for example, adequacy and completeness) of the environment (for example, climatic conditions, equipment, facilities, and measuring devices) in which the verification method and procedures will be implemented.

b) Perform the planned design solution verification using the selected methods and procedures within the established verification environment to

   1) Collect and evaluate verification outcomes to either show conformance to the requirements of the selected physical solution representation or to identify variances (untraceable requirements and constraints, anomalies, variations, voids, and conflicts);

   2) Resolve variances, as appropriate, and reverify to establish compliance, when the cause of the variance was failure to properly complete the fully characterized design.

c) Reverify according to a redesigned verification plan, test method, or procedure when variances were determined to be caused by poor verification or inadequate verification environmental preparation.

d) Record verification results, including: corrective actions taken; lessons learned; outcomes achieved; tradeoff, effectiveness, and risk analyses completed with resulting key decisions; tests activities completed; variances; and the verified design solution in the information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement show that

1) the system design solution appropriately integrates the end products, the enabling products, and the external interfacing products as appropriate to the level of the system structure and enterprise-based life cycle phase;
2) the functional and performance requirements of the selected physical solution representation are satisfied;
3) the functions of the selected physical solution representation have been implemented correctly;
4) the system constraints are satisfied, including physical, functional, and human interfaces.

Requirement 31—End Product Verification

The developer **shall** verify that an end product to be delivered to an acquirer conforms to its specified requirements.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Plan the end product verification in accordance with the Verification Plan, agreement, applicable enterprise-based life cycle phase, and level in the system structure. Include

1) selection and definition of the appropriate method for end product verification;

2) verification procedures to be followed for the method selected, the purpose and objective of each procedure, pretest and post-test actions, and the criteria for determining the success or failure of the procedure;

3) establishment and checkout (for example, adequacy and completeness) of the environment (for example, climatic conditions, equipment, facilities, and measuring devices) in which the verification method and procedures will be implemented;

4) assurance that the test articles are on hand, assembled, or integrated with the verification environment according to verification plans and schedules, and that appropriate sets of specified requirements are available.

b) Verify the end product, using the selected methods and procedures within the established verification environment to

1) collect and evaluate verification outcomes to either show compliance or identify variances (untraceable requirements and constraints, anomalies, variations, voids, and conflicts);

2) for variances not caused by poor test conduct or conditions, complete appropriate tasks of the Planning Process, the Control Process, the Requirements Definition Process, and the Solution Definition Process to resolve variances, and then repeat this set of End Product Verification tasks.

c) Reverify according to a redesigned verification plan, test method, or procedure when variances were determined to be caused by poor verification or inadequate verification environmental preparation.

d) Record verification results, including corrective actions taken; lessons learned; outcomes achieved; tradeoff, effectiveness, and risk analyses completed with resulting key decisions; tests activities completed; variances; and the verified end products in the information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement show that the integrated composite of end products

1) complies to its specified requirements;
2) functions together with other system end products and with interfacing products throughout the performance envelop;
3) is ready for delivery to the acquirer, in accordance with the agreement.

Requirement 32—Enabling Product Readiness

The developer **shall** determine readiness of enabling products for development, production, test, deployment/installation, training, support/maintenance, and retirement or disposal.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Plan enabling product readiness determination and associated process proofing in accordance with the appropriate plan, maturity of the related end products, agreement, applicable enterprise-based life cycle phase, and level in the system structure. Include

1) selection and definition of the appropriate method for the enabling product readiness determination and for proofing for each applicable associated process;

2) readiness determination procedures to be followed for the method selected, the purpose and objective of each procedure, pre-test and post-test actions, and the criteria for determining the success or failure of the procedure;

3) establishment and checkout (for example, adequacy and completeness) of the environment (for example, climatic conditions, equipment, facilities, measuring devices) in which the readiness determination method and procedures will be implemented;

4) assurance that required information regarding the status and maturity of enabling product development or requirements definition is available and that non-developmental enabling products are available and, if appropriate, integrated with the environment according to appropriate plans and schedules.

b) Do the planned enabling product readiness determination and associated process proofing, using the selected methods and procedures within the established environment to

1) collect and evaluate readiness determination outcomes to either show compliance or identify variances (untraceable requirements and constraints, anomalies, variations, voids, and conflicts);

2) for variances not caused by poor readiness determination or process proofing conduct or conditions, complete appropriate tasks of the Planning Process, Control Process, Requirements Definition Process, and Solution Definition Process to resolve variances, and then repeat the readiness determination or proofing.

c) Reaccomplish readiness determination according to redesigned plans, test method, or procedure when variances were determined to be caused by poor readiness or proofing conduct or by inadequate environmental preparation.

d) Record readiness determination and process proofing results, including corrective actions taken; lessons learned; outcomes achieved; tradeoff, effectiveness, and risk analyses completed, with resulting key decisions; tests activities completed; variances; and the verified enabling products and proofing of associated processes in the information database.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement show that

1) associated process requirements for production, test, deployment, training, support, and disposal have been identified;

2) plans and selected methods, procedures, and tools for each associated process will be able to accomplish their intended purpose;

3) enabling product development for each associated process will be completed and enabling products will be available when needed, or satisfactory non-developmental products will be made available, to provide the required support functions to the intended end product;

4) associated processes are proofed properly (for example, proof test of the manufacturing processes for rate production) against requirements and can perform their purpose with respect to support of the intended end product.

NOTE—For each associated process, enabling products requiring development will go through both design solution verification and end product verification as the processes of this Standard are implemented for that development. Off-the-shelf or reused enabling products will be validated against the acquirer requirements, when appropriate. These non-developmental enabling products will be required for verification of physical and functional interfaces with their related end products during the associated end product verification.

4.5.4 End Products Validation Process

The End Products Validation Process is used to demonstrate that the products to be delivered, or that have been delivered, satisfy the validated acquirer requirements (for example, customer, user, or operator requirements, or assigned requirements) that were input to the system design processes and that are applicable to the resulting end products.

Requirement 33—End Products Validation

The developer **shall** ensure that an end product, or an aggregation of end products, conforms to its validated acquirer requirements.

The developer **should** plan and do appropriate tasks to complete this requirement. Tasks to consider include the following:

a) Determine the type of end product validation required and the exit criteria, including the acquirer requirements applicable to the system end products being validated.

NOTES

1 For a system that is an aggregation of end products (see building block discussion in Subclause 6.1), the individual end products and the aggregation of end products are to be validated.

2 The types of end product validation include: (1) validation against validated acquirer requirements in the anticipated usage environment with test conditions that span the expected range of actual operating conditions; (2) certification tests against established certification requirements; (3) acceptance tests, using operational processes and personnel in an operational environment; or (4) as specified in the agreement.

b) Acquire the test article, or aggregation of end products, for the validation as appropriate to the enterprise-based life cycle phase and level of the system structure.

NOTE—The test article is typically the product, or an aggregation of products, that is to be delivered or that has been delivered, and that has already been verified. In early enterprise-based life cycle developments, the product or aggregation of products undergoing validation can be a virtual prototype, breadboard, brassboard, or model. Thus, a detailed simulation, operated so that acquirer perceptions can be evaluated, is a possible means of validation.

c) Conduct the end products validation in accordance with the Validation Plan, as required in the agreement, to show conformance with appropriate requirements; collect and analyze validation outcomes to identify any variances; and do appropriate process tasks to resolve variances and repeat appropriate verifications and validations.

d) Revalidate with improved or correct procedures and equipment, when variances were caused by poor test conduct or conditions.

e) Record the validation outcomes, procedures, assumptions, lessons learned, and other pertinent information about the validation and results to provide traceability.

The expected outcomes for these representative tasks are provided in Annex C. The outcomes associated with completing this requirement provide end products that conform with acquirer requirements stated in an agreement, including any approved changes, or certification or acceptance criteria, as appropriate.

NOTES

1 The key difference between end product validation and end product verification is that end product validation answers the question: Does the delivered end product conform to the validated input acquirer requirements, certification criteria, or acceptance criteria, as applicable? End product verification answers the question: Does the output end product comply to the output specified requirements from which the end product was built, coded, procured, or assembled and integrated?

2 Processes or manual procedures that are part of the defined solution are implicitly included in this validation, since they are a type of product.

3 Requirement 33 addresses the validation of each end product, or aggregation of end products, against validated acquirer requirements. There can be cases where it is also appropriate to validate against other stakeholder requirements.

4 In addition, there can be cases where it is appropriate to validate against actual needs and expectations of end users in their environment under real-world conditions. This is called by various names: market trial, field testing, beta testing, or operational test and evaluation.

5 Application context

This Clause describes the application context for the requirements of this Standard. Figure 5.0 shows external enterprise and project factors that have the potential to affect, or to be affected by, project interfaces.

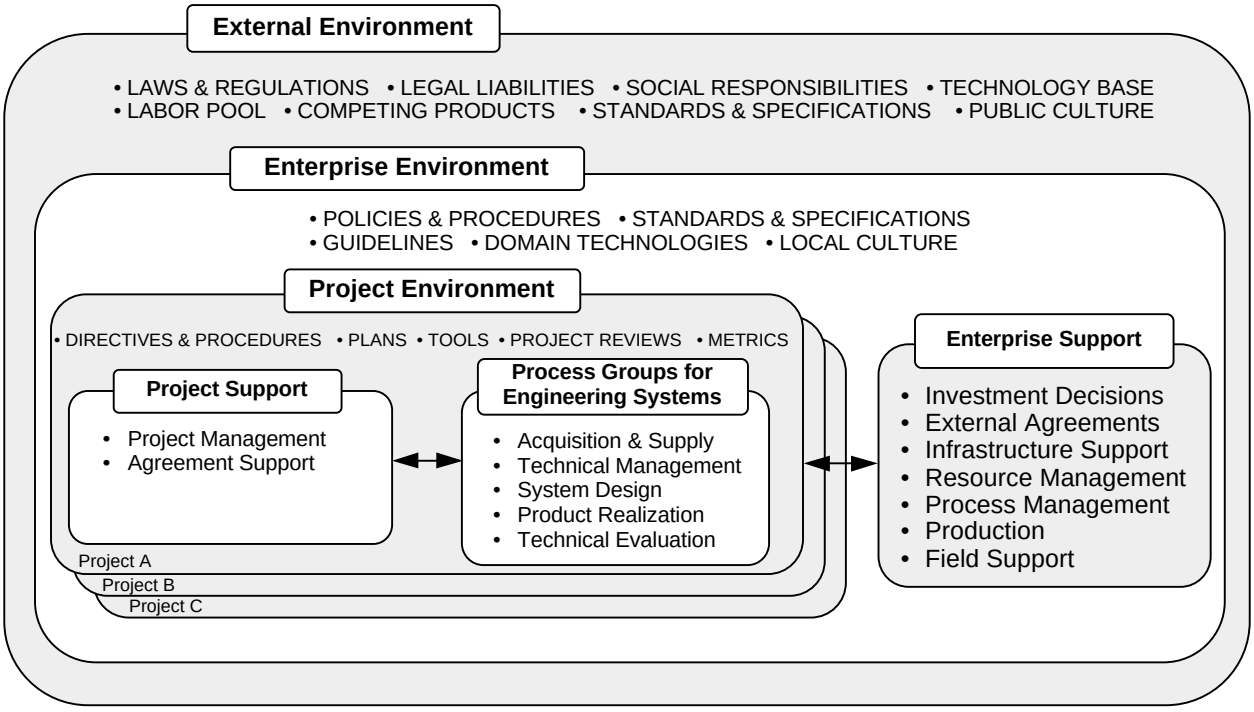

Figure 5.0—Context for application of this Standard

5.1 Enterprise factors

The enterprise is the context in which the process requirements of this Standard are intended to be adopted, directed, and implemented. The enterprise is the source of project start-ups and of project cancellations, and is the source of infrastructure and resource support. Enterprises respond to, as well as create, the markets for system products created by projects within the enterprise. The enterprise further manages the multiple projects within the enterprise to most effectively apply resources and use the infrastructure. The enterprise also establishes constraints of technologies used in existing product lines, as well as manufacturing and test facilities, and support service limitations that constrain project performance.

It is in this context that the enterprise prepares policies and procedures to create or cancel projects, and by which projects perform the processes of this Standard.

5.2 Project factors

5.2.1 Enterprise support

Projects create systems consistent with the business strategy of the enterprise and within the constraint of the enterprise factors cited in Subclause 5.1. Specifically, the following support is to be expected from the enterprise:

a) Investment decision support, including business needs assessments, selection of new start projects, determination of project continuance, and allocation of financial resources for equipment, tools, and training;

b) Agreement support, including contracting, bid and proposal funding, proposal preparation, and oversight (when an external agreement is required);

c) Infrastructure support, including research and development, marketing, facilities, in-service support, computer services, and other services that enable the project to meet its obligations;

d) Resource management support, including financial management, personnel management, training and education of project personnel, office and computer equipment, maintenance, and shipping;

e) Process management support, including establishment of standard procurement processes and methods, guidelines for tailoring adopted processes from this Standard, selection and acquisition of tools, assessment of directed process implementation and monitoring of process effectiveness, and improvement of processes;

f) Production support, including fabrication, construction, manufacturing capacity, and staffing; equipment and tools; and accomplishing fabrication, construction, manufacturing, quality control, and testing;

g) In-service support, including installation, customer support, product upgrades, warranty service, field modifications, on-site consulting, and product certification.

The availability and adequacy of enterprise support functions determine the viability of a project, schedule of project tasks, capability to satisfy an established agreement with another enterprise, and the availability of personnel who have the skills and knowledge to complete project responsibilities.

5.2.2 Project support of the technical processes

Projects provide the context in which a system is engineered. Projects use the processes from this Standard as directed by enterprise policies, or as directly adopted by the project, to satisfy agreements. Directives and procedures are prepared by the project to guide both the project management functions and the technical efforts applicable to the specific project. In this context, the technical efforts to meet the requirements of this Standard require project functional support. Such support includes;

a) Agreement support including preparing appropriate tasking agreements between projects, or within the project, to implement the planned technical effort, and providing proposal preparation support, as applicable;

b) Project management including project integration, scope management, time management, cost management, quality management, human resource management, communications management, risk management, and procurement management.

NOTE—More information on the types of support to be expected is in *A Guide to the Project Management Body of Knowledge* (PMBOK) published by the Project Management Institute.

The availability and adequacy of these project functions, and the project directives and procedures, determine the tasks and scope of the processes for engineering a system. The enterprise determines the tools, equipment, and metrics to be used, and the reporting and management review requirements.

5.3 External factors

The external environmental factors that can affect the processes for engineering a system include local, state, national, and international laws and regulations; potential legal liabilities; social responsibilities; available technologies; the labor pool; competing products and technologies; and national or international standards and specifications. Also, the processes for engineering a system can be affected by external agreements for upper or lower development projects and requisitioned end products and by existing external infrastructures and the physical world.

Systems and their products operate with organizations and personnel who use the products, and with other operational entities that provide input to the system, or otherwise interact with the system products, but are not part of the system under development and are not controlled by the developer. The interaction and interfaces (physical or functional) between the system products and their external operational environment can affect the implementation of the processes used for engineering the project system. Changes in the operational environment can strongly affect system effectiveness and functionality. System performance and adequacy also can be affected by the system's ability to respond both to the operational environment and to changes in that environment.

5.4 Influence of other enterprise projects

Enterprises often have more than one development project at once. Two such projects can sometimes benefit from the exchange of products, for example, parts, subassemblies, or data. Agreements between such projects are established, as appropriate.

6 Application key concepts

This Clause describes key concepts for application of the processes of Clause 4 to the engineering or reengineering of a system. There are two aspects to this clause: first, the system to which the processes are applied; and second, the top-down development of system products and the bottom-up implementation and realization of system products. The first is the basis for the system structure; the second is the basis of an engineering life cycle.

6.1 System concept

The system to which the processes of Clause 4 are applied consists of both the end products to be used by an acquirer for an intended purpose and the set of enabling products that enable the creation, realization, and use of an end product, or an aggregation of end products. Enabling products are used to perform the associated process functions of the system—develop, produce, test, deploy, and support the end products; train the operators and maintenance staff of the end products; and retire or dispose of end products that are no longer viable for use. Both the end products and the enabling products are either developed or reused, as appropriate. The relationship of these system elements is shown in Figure 6.1.

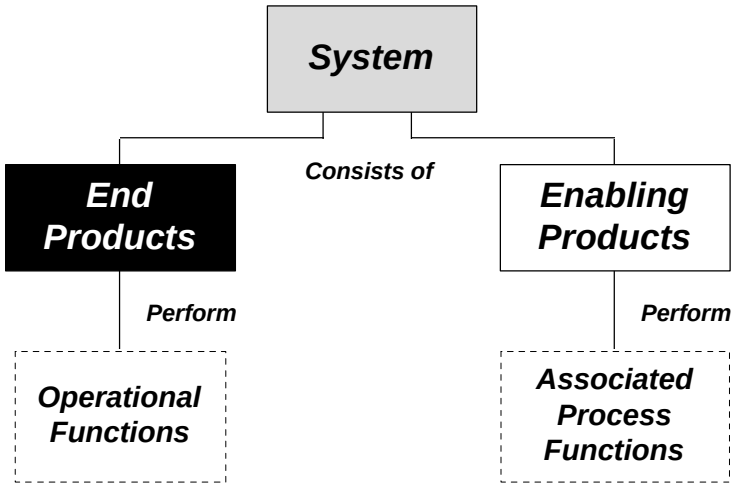

Figure 6.1—System concept

NOTE—The above system concept implicitly includes the personnel who develop, produce, test, operate, support, and retire the system products, as well as both those who train others involved with these system functions, and the human factors issues and concerns associated with these personnel. Such personnel and human factors issues are included in the applications of the processes of this Standard to the building block structure derived from this system concept.

6.1.1 Building block

The system forms the basis for a larger structure, called the *building block*, shown in Figure 6.1.1. The building block provides the framework for application of the processes of Clause 4.

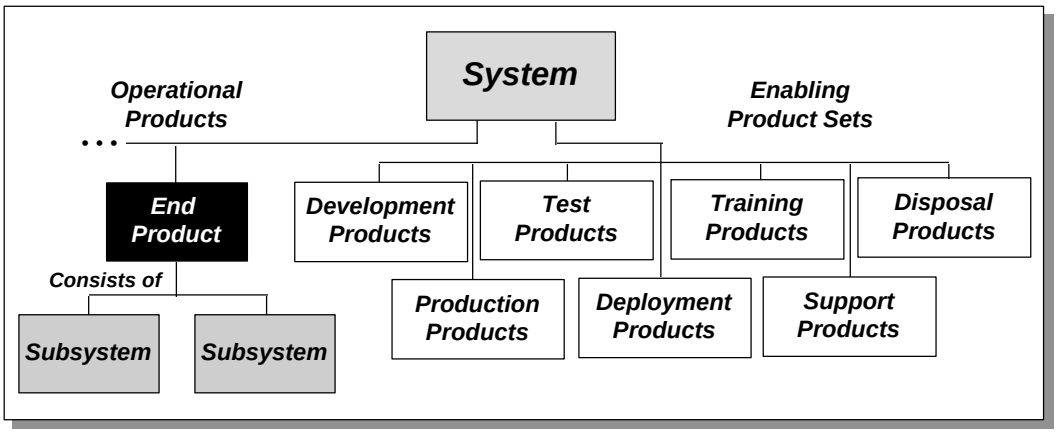

Figure 6.1.1—Building block

A building block is made up of the *system* (gray element), one or more end products (black elements), two or more *subsystems* (gray elements) for each end product, and the ensemble of *enabling products* (white elements). Each end product and each enabling product includes one or more of the following: hardware, software, firmware, personnel, facilities, data, materials, services, and processes. The following information can be associated with each element within the building block:

a) configuration identification;

b) the costs to be collected;

c) identifications of interfacing elements inside and outside the building block;

d) specifications relevant to the element;

e) definition of work to be done;

f) other relevant agreement information.

6.1.1.1 System element

The *system* element of the building block is the object for which the developer defines the acquirer and other stakeholder requirements using the Requirements Definition Process.

6.1.1.2 End product elements

The *end products* perform the operational functions for the system. These products are developed using the Solution Definition Process, are verified against the specified requirements using the System Verification Process, and are validated against acquirer requirements using the End Products Validation Process.

An end product can be either a legacy product that is being reengineered or a product that the enterprise both has the expertise to make and has similar products already in the market place. Such developments are identified as precedented, derivative, or next-generation. When the specific end product is not *a priori* known, or when the enterprise has limited experience in development of a new system, the development is identified as unprecedented or as a new concept. The application of the System Design Processes to each of these product types is provided in Annex F.

NOTE—Suppose it is already known that a radio set, a radar, an automobile, or another specific product (including acquirer-furnished equipment) is to be used as an end product. Even though the product type is known (precedented), the specific solution for this next-generation product can be defined using the processes of this Standard to satisfy acquirer requirements.

An end product can be self-contained in terms of its use and operations. It also can be an item that has no use outside a larger end product, but that is developed as an end product of a subsystem (lower-layer system building block) using the System Design Processes.

NOTES

1 Examples of self-contained end products are an aircraft, an automobile, a communications satellite, a nuclear reactor, a telecommunication switching module, or a space vehicle that is delivered to an operator.

2 An end product could also be any of many products that make up a self-contained end product. Examples of such end products are an engine or a radio for an aircraft, a power train or a brake for an automobile, a solar panel or a transmitter for a satellite, a control panel or a control valve for a nuclear reactor, a switch or a transducer for a telecommunication switching module, or a life support package or a hatch door for a space vehicle. Such end products can be found at the assembly, subassembly, line replaceable unit, component, or part levels of a system.

3 The end product element is black to represent those elements of the building block that are physically integrated with end products of upper- and lower-layer building blocks to form a composite end product and eventually a self-contained end product.

4 There can be more than one end product in a building block. In such cases, the system consists of an *aggregation* of end products, plus their enabling products.

6.1.1.3 Subsystem elements

If end products cannot be manufactured or are not off-the-shelf products that can be reused or purchased from another supplier, subsystems of an end product are developed using the processes of this Standard. Each end product that is developed consists of two or more subsystems (gray elements). When a subsystem is developed, another lower-layer building block is established (see Subclause 6.2). The hierarchy of such building blocks is called the *system structure*.

6.1.1.4 Enabling product elements

Enabling products perform the associated process or non-operational functions of the system. The enabling products are verified to be ready to perform their intended functions when required to support their related end product, or aggregation of end products. When each set of enabling products is developed using the processes of this Standard, another building block is formed (see Annex G, Figure G.3). Development of an enabling product building block is normally initiated after the related end products are fully defined and after the requirements for enabling products are identified. The building block structure for an associated process is related to only its parent system building block and does not infer development of all products related to an associated process for the entire system structure (upward and downward in the hierarchy of building blocks).

NOTE—Application of the processes of this standard to a building block establishes the specified requirements for the items represented by black or gray elements of Figure 6.1.1. However, only requirements (which most often are not valid technical statements) are initially identified and collected for the enabling products. To represent this difference, the enabling product elements are shown in white. The processes of this Standard are then applied to each set of enabling products to obtain validated technical requirements and, ultimately, derived requirements and specified requirements for these enabling products.

Examples of enabling products developed in conjunction with a system are listed in Table 6.1.1.4

Table 6.1.1.4—Examples of enabling products for each associated process

Associated process	Examples of enabling products
Development	Development plans and schedules, engineering policies and procedures, integration plans and procedures, information database, automated tools, analytical models, physical models, engineering and management personnel, and connecting cables and other interface structures not being developed as separate end products.
Production	Production plans and schedules, manufacturing policies and procedures, manufacturing facilities, jigs,

6.1.2.1 Specifications

Specifications document the specified requirements that are an output from the Solution Definition Process. The building block relationships of black and gray element specifications and the white element requirements, as well as appropriate interface specifications, are shown in Figure 6.1.2.1.

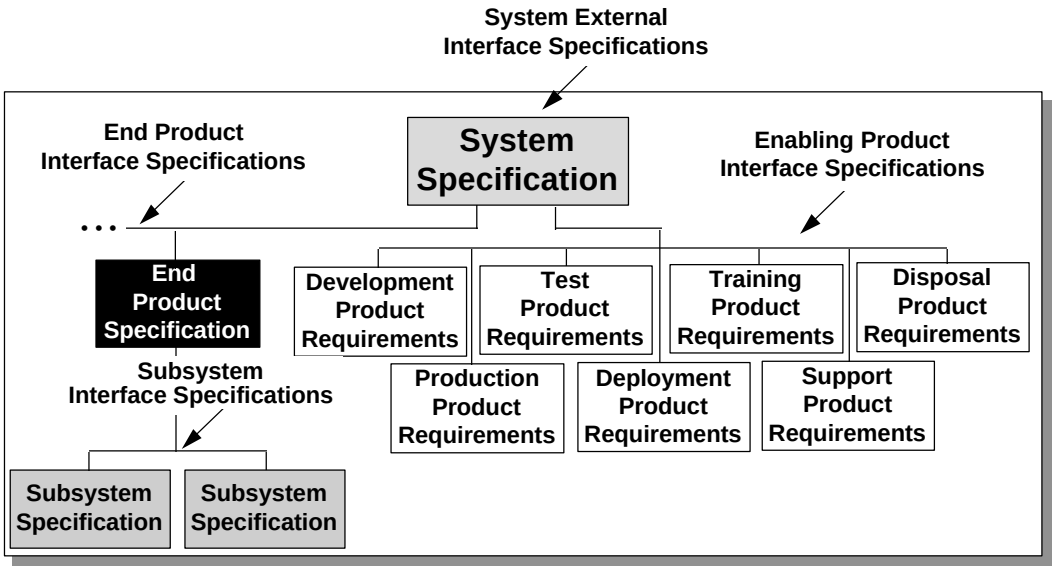

Figure 6.1.2.1—Building block role - specifications

NOTES

1 The gray elements represent system elements that will be defined by specifications produced by application of the System Design Processes (see Subclause 4.3), but are not delivered as a unit.

2 The system specifications are the basis of developing end product specified requirements and associated process requirements. Each subsystem specification is the basis for development of the next lower-layer building block (see Subclause 6.2).

Specifications describe the required characteristics of end products (black elements) or a group of products (gray elements). Characteristics include

a) the functional and performance requirements;

b) interface requirements;

c) the environments in which the product(s) is required to perform its functions;

d) physical characteristics and attributes;

e) the basis for evaluating test articles;

f) the methods for verifying compliance;

g) the intended uses;

h) enabling product requirements.

6.1.2.1.1 Stages of maturity

The specifications for the system, end product, and subsystem elements evolve through three stages: *conceptual*, *initial*, and *established*. *Conceptual specifications* are used to show feasibility of a higher-level initial specification (e.g., end product) and to record the characteristics of notional products. Conceptual specifications are evolved to initial specifications by application of System Design Processes. *Initial specifications* are used to direct lower-layer building block developments of subsystems. The initial specifications evolve into established specifications by application of the System Design Processes. *Established specifications*

a) enable making valid estimates of work and resources needed for the next lower-layer building block development;

b) provide a basis of communication with and among the development team, suppliers, and customers;

c) provide guidance to testers for completing System Verification and End Products Validation Processes;

d) provide a basis for negotiation of engineering changes;

e) guide preparation of detailed drawing or software development file design definitions;

f) enable development of lower-layer building block specifications and solution definitions, e.g., drawings, parts lists, and code lists;

g) enable configuration management (control and maintenance) of solution definitions that satisfy technical requirements;

h) enable the definition of logistics support for spares, replacement parts, training, manuals, maintenance operations, diagnostic tools, and support equipment.

6.1.2.1.2 Performance specifications

Performance specifications are used when it is appropriate to state requirements in terms of

a) the required results without stating the method for achieving the required results;

b) function (what is to be accomplished) and performance (how well each function is to be performed);

c) the environment in which the product(s) must perform these functions;

d) the interface and interchangeability characteristics;

e) the means for verifying compliance.

6.1.2.1.3 Detail specifications

Detail specifications are used when it is appropriate to state design requirements in terms of

a) material to be used;

b) how a requirement is to be achieved;

c) how a product is to be fabricated or constructed.

NOTE—Detail specifications are applicable to guide creation of detailed drawings or the software development file: pseudocode and software dictionary.

6.1.2.2 Interface definition

Interface specifications are essential in most system development activities to clarify interdependencies between system elements within the building block (internal) and other systems above, below, and at the same layer of development (external). Interface specifications are used to define and specify

a) physical and functional relationships between system elements, including operators;

b) functional requirements resulting from these relationships;

c) performance requirements resulting from these relationships;

d) constraints.

6.1.2.3 Multidisciplinary teamwork

Another role for the building block is to enable multidisciplinary teamwork. A reference structure for team assignment is shown in Figure 6.1.2.3. Teams themselves do not ensure teamwork. It is how the teams are integrated that is important, as well as the assignment of properly skilled team members. A system core team is usually composed of the project technical manager, along with members to be assigned to team lead positions on end-product and associated process teams. An end-product team can be the leaders from their respective subsystem team. An enabling product team can be individuals representing their respective functional disciplines. These functional specialists are also assigned to subsystem teams, as appropriate. A subsystem team is normally appropriate domain experts as well as functional specialists and other required specialists. A subsystem team becomes the core team for the next lower-layer building block development of subsystem end products.

NOTE—As with the application of any complex process, training of all team members in the application of the concepts and practices in this Standard is key to its successful application. Successful training includes both training that brings new team members up to speed and training that refreshes existing team members on the currently active elements of the process as the project proceeds.

Multidisciplinary teamwork ensures the accuracy and completeness of the evolving technical data package from which test articles, pre-production prototypes, and production products are to be manufactured or coded.

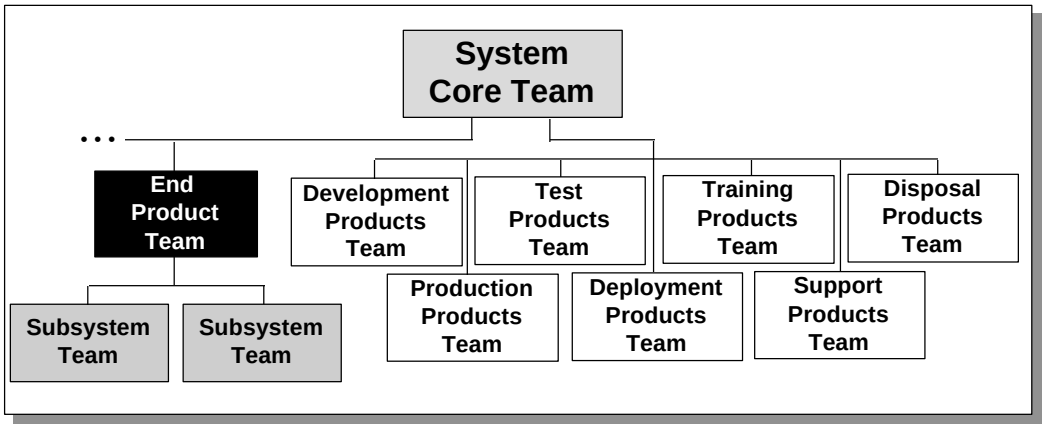

Figure 6.1.2.3—Building block role - teamwork

6.1.2.4 Risk management

Another role for the building block is to provide a structure for assessing and managing risks. The risk associated with arriving at the solution definition for each end product is a function of the risk assigned to each subsystem of the end product. Likewise, the risk associated with the system of the building block development is a function of the end-product risks and the associated enabling-product risks. The building block also shows the relationships between subsystems and end products, and between associated end products that must be considered in determining the risk associated with each end product development. Based on the degree of risk and the relationship among building block elements, risk aversion plans are created and tracked.

NOTES

1 Risk depends on the probability of occurrence and its consequences. Risk is potential harm to the project or system under development. Risk is assessed for project, product, and process aspects of the system. This includes the adverse consequences of process variability. The sources of risk include: technical (for example, feasibility, operability, producibility, testability, and systems effectiveness); cost (for example, estimates and goals); schedule (for example, technology/material availability, technical achievements, and milestones); and programmatic (for example, resources).

2 Risk management requires discipline. Risk management is useful only to the degree that it highlights the need for action, and that action leads to the problem being addressed quickly and thoroughly. Moreover, risk management is continuous. Things can go wrong until the last phase of the project is completed.

6.1.2.5 Technical reviews

Technical reviews are scheduled and conducted during each engineering life cycle phase, as appropriate, to review progress against plan, against the established agreement, and against the applicable enterprise-based life cycle phase exit criteria. They are conducted to determine whether to continue the investment in future engineering or enterprise-based life cycle phases, based on

a) the risks and costs associated with lower-layer developments;

b) the maturity of the development to date;

c) if requirements and technical plans being tracked are on schedule and are achievable within existing project constraints;

d) resources required for lower-layer projects;

e) readiness to proceed, to include external supplier availability and agreement preparations, if applicable.

The building block also is a convenient framework for technical reviews called out in the agreement or the engineering plan. Two types of reviews are conducted—*Incremental* and *System*. *Incremental Reviews* are conducted on subsystems, associated processes for related sets of enabling products, and end products. Upon completion of the incremental reviews, a *System Review* (top element of the building block) is conducted. The typical order of these reviews is shown in Figure 6.1.2.5.

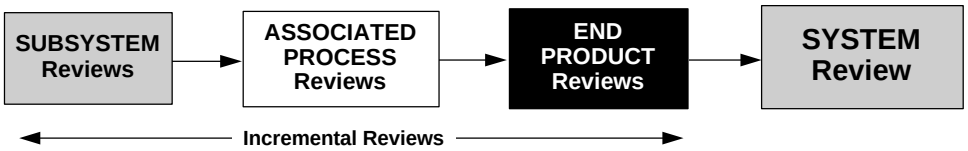

Figure 6.1.2.5—Building block role - technical reviews

The conduct, the reviewing body, and the presenters of specific technical reviews are planned in a technical review plan during the Planning Process. The team associated with a specific review is assigned the task of creating and presenting the technical review. For *Subsystem Reviews,* the parent end product team is typically the reviewing body. End product team members and team leads selected from other associated process teams make up the reviewing body for the *Associated Process Reviews*. These reviews can be held as a joint review. The core team is the reviewing body for *End Product Reviews*. Reviewing bodies can be supplemented by other specialists from outside the project, as appropriate to meet technical review objectives. The reviewing body for a *System Review* can be designated in the agreement and/or in the project plan or engineering plan. The *System Review* can be held along with a project review when intended to meet exit criteria for an enterprise-based life cycle phase.

The purposes of the Incremental and System Reviews are listed in Table 6.1.2.5.

Table 6.1.2.5—Purposes of technical reviews

Review	Purpose
Subsystem	To assess progress in defining and satisfying subsystem requirements.
Associated Process	1) To assess progress and identify issues associated with requirements for one associated process or group of associated processes; 2) to ensure the suitability and availability of the services of enabling products when they are needed.
End Product	To address issues and demonstrate required building block development progress and maturity.
System	See Annex E.

The technical reviews applicable for the engineering life cycle (see Subclause 6.3) are described in Annex E. The incremental reviews are to be completed prior to each Annex E system technical review.

6.1.2.6 Cost collection and reporting

Another use of the building block structure is for collecting and reporting costs related to engineering life cycle activities. The costs are incurred in each building block system element as development activities are done in accordance with assigned work packages generated during planning. The costs incurred include direct labor costs associated with applying engineering process tasks for requirements definition, design definition, design verification, tradeoff and effectiveness analyses, fabrication, software bulk copying, technical reviews, data and document generation, integration, and testing.

Technical agreement, planning, and control costs are also collected and reported as part of the development of associated process enabling products. The costs associated with a building block system development can be easily summarized by rolling up the costs of subsystems, end products, and associated processes. When the project performance is tracked by an acquirer, or for internal control, using a cost performance measurement system, cost and performance measurements are combined using an earned-value approach.

6.2 System structure concept

A single building block rarely defines the complete solution to acquirer and other stakeholder requirements. If a subsystem requires further development, this is done as a subordinate building block development. Lower-layer building block developments are initiated as soon as definite contents of the building block are determined. The definite contents of the building block are represented as end product established specifications, initial subsystem specifications, interface specifications, and requirements identified for applicable enabling products of the associated processes. Building blocks are connected to form the system structure, or a building block hierarchy. The relationship among building blocks in a hierarchy is shown in Figure 6.2.

This layered approach in the decomposition of building blocks continues until: 1) the end products of a building block can be implemented; 2) the requirements for an end product can be satisfied by an existing product; or 3) the end products can be acquired from a supplier. The specific building block structure will vary with each system, based on the number of end products, the number of subsystems in an end product, and the applicable enabling products of the associated processes.

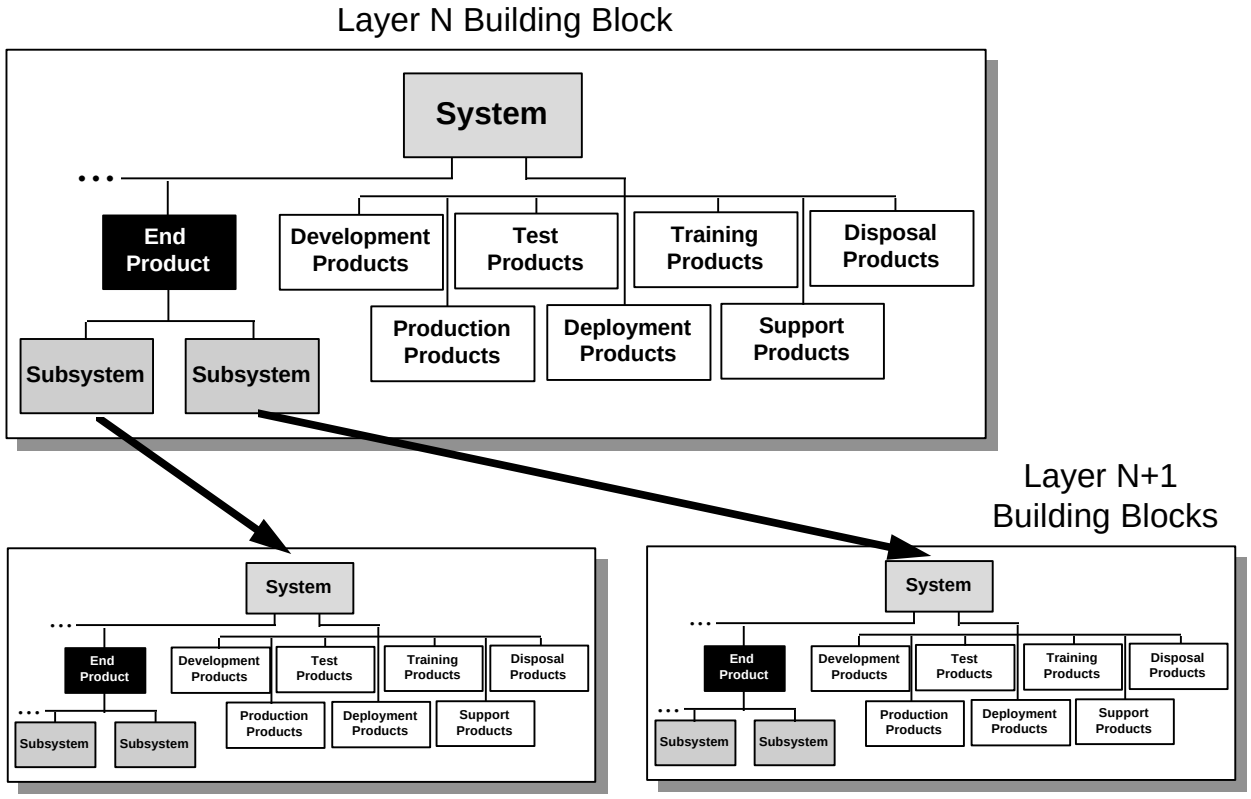

Figure 6.2—Forming a system structure

NOTE—A system structure serves as the framework for the engineering of a system. Although represented in Figure 6.2 as a one-to-one decomposition, some cases can occur that have multiple inheritances when the same subsystem or end product can be used several places in the system structure.

The *specified requirements* for a subsystem become the *assigned requirements*

6.2.1 Top-down development

Figure 6.2 shows a view of the layered development approach for a project (also known as a program in some domains). Typically, the project receives acquirer requirements in a formal agreement (see Subclause 4.1) and provides reports and delivers products in accordance with that agreement (see Subclause 4.2). Each project can have several lower-layer building block developments. An agreement is used for each lower-layer building block development using requirements assigned from the parent upper-layer building block. Typically, only one engineering plan (see Subclause 4.2) is required for the multiple layers of building block developments within a single project. If an external supplier is used for a lower-layer building block development, a formal agreement is required.

Figure 6.2.1a is an example system structure showing a layered development. The top building block contains the end product that must satisfy the primary user's or customer's requirements. This top building block represents what is often called the prime contractor's project. Two other projects are shown: Project A and Project B. The top building block in each of these projects represents the top layer of development for the respective project, but the second layer for the prime contractor's project. Project A spawns two layers of development, whereas Project B spawns multiple lower-layer building block developments. The lines connecting the layers reflect the specified requirements assigned from a parent building block to its subordinate building block.

NOTES

1 It is recognized that three approaches are practiced to engineer a system—top-down, bottom-up, and middle-out. The approach in this Standard could be considered both middle-out and top-down. Since the hierarchy of building blocks of Subclause 6.2 starts in a project that could be anywhere in the system structure, this could be considered middle-out.

2 The top-down approach is intended to flow-down requirements so as to ensure satisfaction of top-layer building block project customer requirements. It is also intended to take advantage of reuse and off-the-shelf items that satisfy assigned requirements in order to lessen development costs and shorten development cycle time. The requirements of this Standard are based on the top-down approach.

3 A bottom-up approach to development is normally not to be used unless it is ascertained that the requirements of the top-layer building block project system are not affected adversely.

A project applies the System Design Processes (see Subclause 4.3) to each building block in the project boundary to develop the appropriate system, end product, and subsystem development specifications that are defined to satisfy assigned and other stakeholder requirements related to a single building block. The end products defined at Project A's second layer of development are both satisfied by off-the-shelf or reuse products and, therefore, do not require further development. Project B's second layer of development has one building block that requires a third layer of development, whereas the specifications of the other building block's end product are satisfied by either an off-the-shelf product or a reuse product. Project B

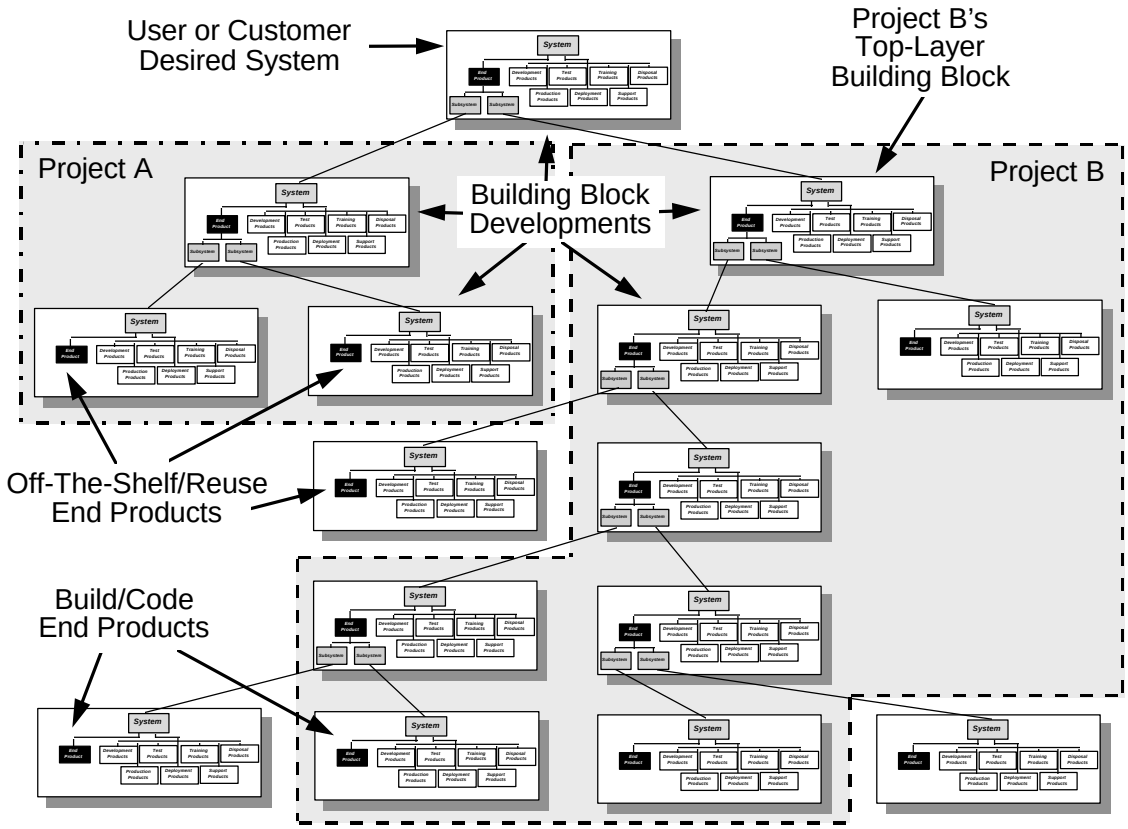

Figure 6.2.1a—Example system structure

Figure 6.2.1b shows, for Project B of Figure 6.2.1a, top-down development using the System Design Processes (see Subclause 4.3). The inputs to each building block include the assigned requirements from the building block above and the other stakeholder requirements that will influence the building block development. The completion of the applicable planned technical efforts on each building block is to result in a set of end product specified requirements and subsystem initial specifications, when further development efforts are required.

The end product specified requirements will be used for End Product Verification, as well as for procurement of off-the-shelf or reuse end products, for building, or for assembly and integration, as applicable. As the technical efforts proceed, design feedback is provided to the parent building block to ensure interface compliance and also to ensure that design decisions do not adversely affect the parent building block end and enabling products, or other subsystems. Likewise, the parent building block provides any changes to requirements that result from other subsystem developments, enabling product developments, or stakeholder changes. Changes are passed downward to lower-layer building block developments, as applicable.

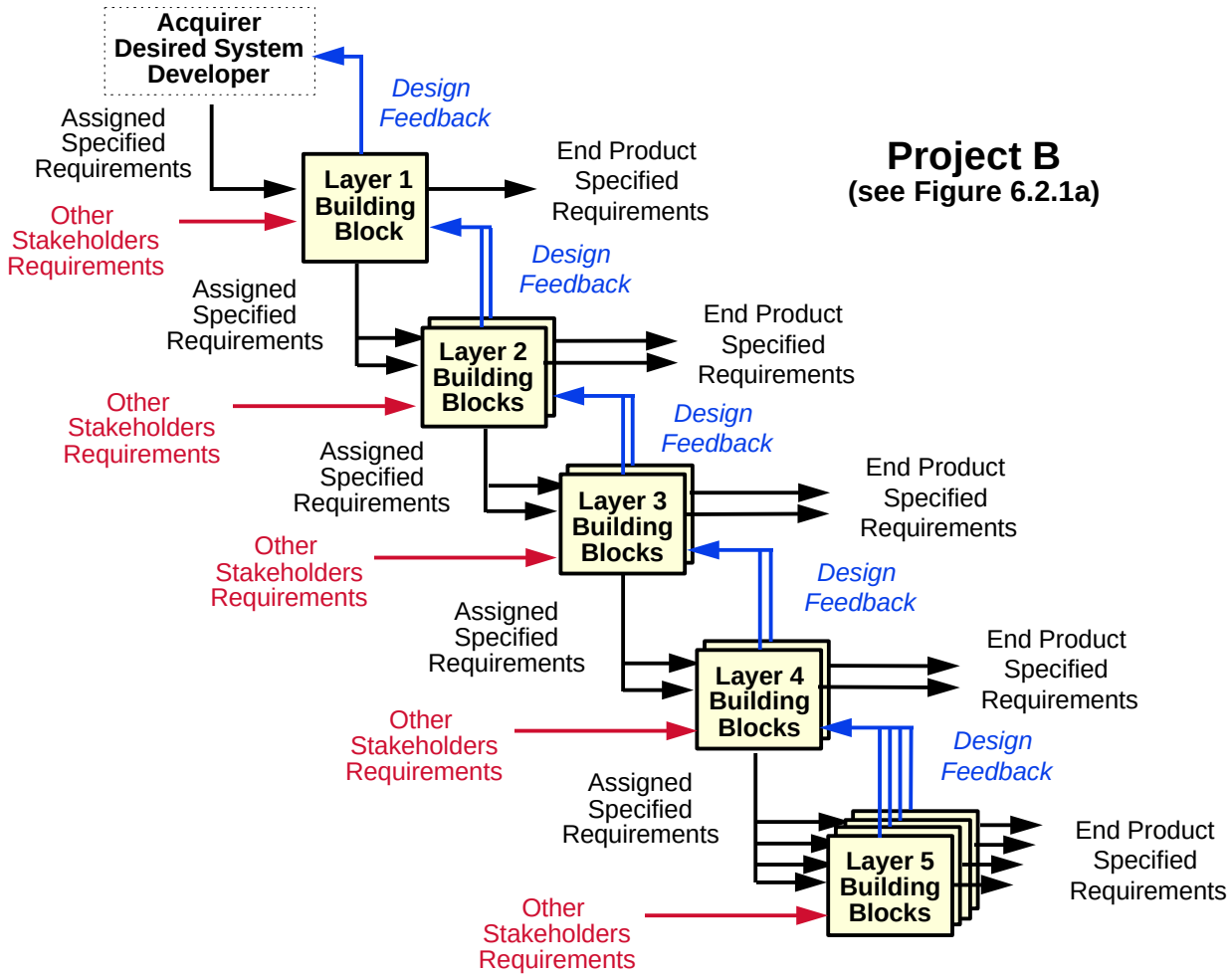

Figure 6.2.1b—Top-down development

6.2.2 Bottom-up realization

The previous subclause explained how end products that make up the system structure are developed, from the top down. Once specific end products are defined sufficiently by specifications so that an off-the-shelf product or a reuse product can be used, or so that the end products can be built or coded, Product Realization Processes (see Subclause 4.4) can be initiated. As was shown in Figure 6.2.1a, this can occur at any layer of the system structure. However, the assembly or integration, verification, and validation of such products occur from the bottom up.

The bottom-up realization of end products is shown in Figure 6.2.2, again for Project B (reference Figure 6.2.1a). The end products procured for layer 5 (built, coded, used off-the-shelf, reused, or delivered by external supplier) are verified using the End Products Validation Process (see Subclause 4.5). Once verified, the end products are delivered, along with verification data, to the parent building block, in accordance with the established agreement. The end product is validated against its assigned requirements, either before delivery by the end product developer or supplier, or by the layer 4 building block developer. Validation is completed using the End Products Validation Process (see Subclause 4.5) before being assembled or integrated with the other validated end products that make up the appropriate composite end product for the layer 4 building block. This composite end product is then verified, and the procedure is

repeated until the project's end product for the layer 1 building block is delivered to the top-level developer in accordance with the agreement.

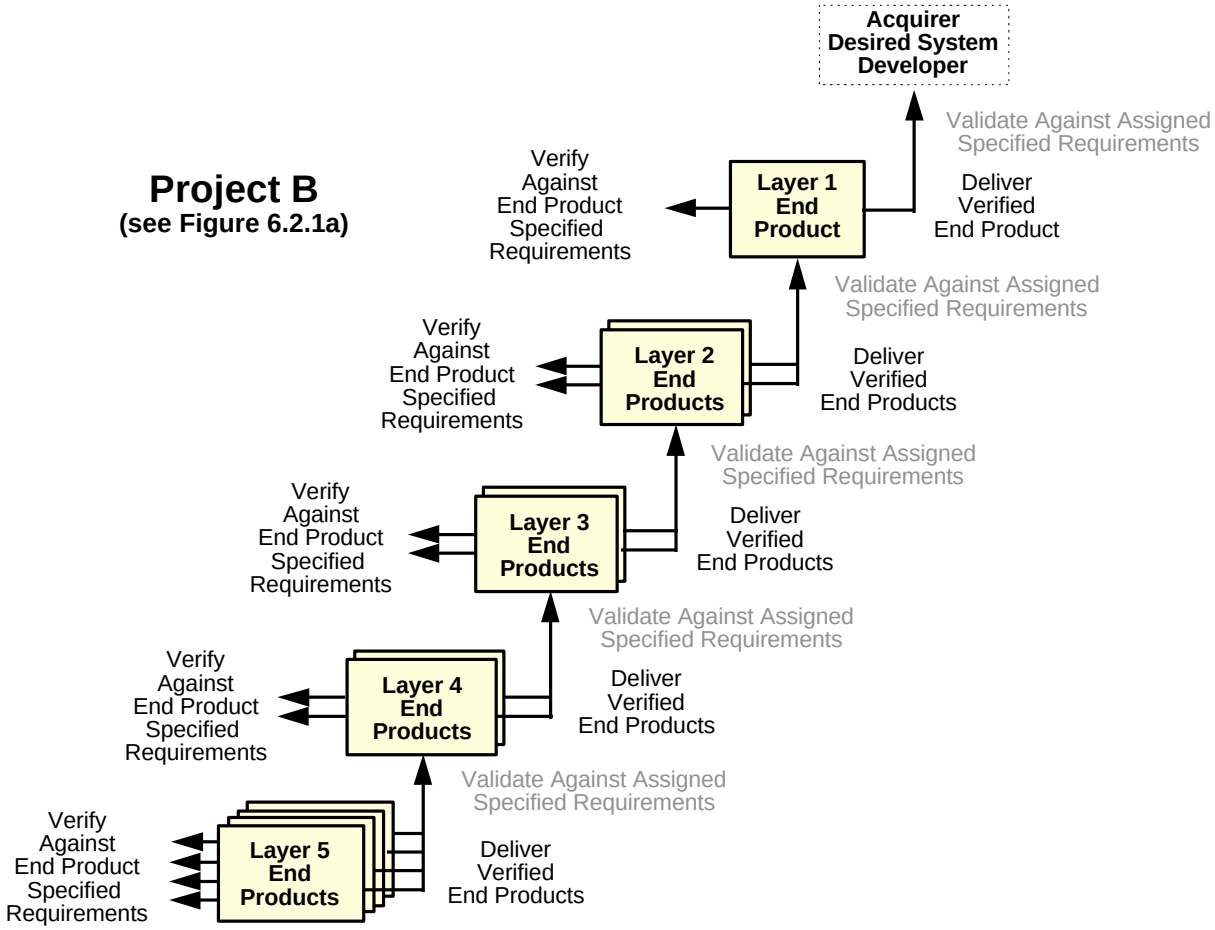

Figure 6.2.2—Bottom-up realization

A key purpose of this bottom-up approach is to discover test-article variances and design anomalies at the lowest layer of development possible in order to prevent lower-layer end product defects from being buried or overlooked and then showing up during top-layer end product verification and end product, or aggregation of end products, validation. The System Design Processes are applied to the affected building block developments to correct anomalies uncovered by System Verification or End Products Validation Processes (see Subclause 4.5). End products that do not comply with specified requirements must be re-manufactured, re-coded, or re-procured to correct the anomaly or deficiency and so that a corrected test article can be verified.

NOTE—End product validation against acquirer requirements can be accomplished before delivery, but after end product verification is complete, if called for in the agreement. Otherwise, the acquirer validates the delivered end product prior to assembly or integration with other end products to make up the composite end products appropriate to the building block. The aggregation of end products might also need to be validated.

6.3 Engineering life cycle concept

Each product within a system structure has its own life cycle. The product line it represents is developed and produced to meet acquirer requirements and is then inserted into the marketplace either to satisfy an established agreement or through marketing. Following insertion, there is a growth stage where the product becomes a viable product in the marketplace. This is followed by a maturity stage where the product is no longer in growing demand, where competitor products take part of the market, or where the product fails to achieve its market potential and demand levels off and starts to decline. Finally, the sales of the product decline, and the product is phased out and is no longer marketed or distributed. All products undergo this life cycle.

NOTE—The product life cycle is the one generally defined in project management books. This life cycle is the driver for the two other life cycles, in that new products must be developed, or that legacy products are improved, to create new business and profitability to an enterprise, or to keep systems competitive to meet external threats to an enterprise or nation.

The processes of this Standard are applicable at any point in a product's life cycle. In the early stages of a product's life cycle, the processes for engineering a system are applied to bring the system, or a portion thereof, into realization. System products are then produced and transitioned into operations where products are used and supported and during which operators and maintainers are trained. As products are used and as design anomalies or desired product improvements are identified, the processes of this Standard are applied to reengineer the products. Finally, during product retirement, the processes of this Standard are applied to correct any enabling product design anomaly for the retirement or disposal process.

The layers of development shown in Figures 6.2.1a and 6.2.2 are directly correlated with a set of engineering life cycle phases during which the processes of this Standard are applied. The engineering life cycle phases are described in Table 6.3. These phases are grouped as follows: (1) *Conception,* consisting of the Pre-System Definition Phase; (2) *Creation,* consisting of the System Definition, Subsystem Design, and Detailed Design phases; and (3) *Realization,* consisting of the End Product Physical Integration, Test, and Evaluation phase.

Figure 6.2.1a shows application of the phases related to Conception and Creation for the top-down development. Figure 6.2.2 shows application of the phase related to bottom-up Realization. Annex B describes how these groups of phases are used in individual enterprise-based life cycle phases to incrementally evolve the system products before implementing the utilization phases of the enterprise-based life cycle, or before continuing the utilization enterprise-based life cycle phase in which the system products were improved using the activities of the engineering life cycle phases.

NOTE—The engineering life cycle applies in the research and development stage of a product's life cycle, but it also applies during any product life cycle phase or enterprise-based life cycle phase when it is needed as a result of engineering or reengineering decisions. (see Annex B)

Table 6.3—Engineering life cycle phases

PHASE	DESCRIPTION OF ACTIVITIES
Pre-System Definition	This is the start-up phase of the engineering life cycle. The Technical Management Processes, as applicable, are applied to plan a technical effort, or refine the technical effort described by existing plans, that is consistent with an established agreement System Design Processes are applied, as appropriate, to the top-layer building block of a project to determine the best system concepts to satisfy acquirer requirements, or to refine a previously selected concept, or legacy system, established in a prior enterprise-based life cycle phase. A set of initial specifications for the system and selected end products of the system concept is defined, as appropriate, and technology requirements, risks, and other constraints are identified. Before progressing to the next phase, appropriate incremental technical reviews and a system concept review are completed.
System Definition	System Design Processes, and appropriate Technical Management and Technical Evaluation Processes, are applied to the top-layer building block of a project to establish specified requirements for the end products and to define initial specifications, including interface specifications, for subsystems of each end product, and to identify enabling product requirements to enable an end product to meet functionality requirements during development, production, test, deployment, training, support, and disposal, as applicable. Identified high technical risk areas are mitigated during this phase. Before progressing to the subsystem design phase, appropriate incremental technical reviews and a system definition review are completed.
Subsystem Design	System Design Processes, and appropriate Technical Management and Technical Evaluation Processes, are applied to the building blocks at the second layer of the project to establish specified requirements for the end products and to define initial specifications, including interface specifications, for subsystems of each end product that requires further development, and to identify enabling product requirements to enable an end product to meet functionality requirements during development, production, test, deployment, training, support, and disposal, as applicable. Identified high technical risk areas for subsystem end products and enabling products are averted during this phase. Before progressing to the detailed design phase, appropriate incremental technical reviews and a system preliminary design review are completed.
Detailed Design	System Design Processes, and appropriate Technical Management and Technical Evaluation Processes, are applied to the building blocks at the third and lower layers of the project to establish specified requirements and detailed drawings or documents, as appropriate, for the end products and to define initial specifications, including interface specifications, for subsystems of each end product that requires further development, and to identify enabling product requirements to enable an end product to meet functionality requirements during development, production, test, deployment, training, support, and disposal, as applicable. Identified high technical risk areas for lower-layer end products and enabling products are averted during this phase. Before progressing to the next lower-layer detailed design effort, appropriate incremental technical reviews and a system detailed design review are completed on the applicable building block elements. When an end product design can be fulfilled by buying, building, or reuse, development of that end product is complete. Prior to progressing to the next phase of the engineering life cycle, test readiness and production readiness technical reviews are completed.
End Product Physical Integration, Test and Evaluation	End products are obtained from suppliers, acquirers (in the case of customer-furnished items), are off-the-shelf, or are fabricated, based on completed detailed design specifications, documents, or drawings. The Implementation Process, Technical Management Processes, and Technical Evaluation Processes are applied to validate end products obtained, to assemble or integrate validated end products, and to verify that composite end products satisfy specified requirements. The Transition to Use Process is applied to deliver the verified end products to the acquirer of the next layer up in accordance with the established agreement. Then, the Implementation Process, the Technical Management Processes, the Technical Evaluation Processes, and the Transition to Use Process are applied, as appropriate, to successive upper-layer building blocks until delivery of the end products and enabling products required in the agreement that established the project.

Annex A—Glossary (normative)

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

acquirer: An enterprise, organization, or individual that obtains a product (good or service) from a supplier.

NOTES

1 The acquirer can be a customer or user of a desired system product, or can be a developer obtaining a lower layer product in the system hierarchy from another vendor or a developer in the role of supplier.

2 An acquirer is a type of stakeholder.

agreement: An arrangement, not necessarily contractual, between two parties (an acquirer and a supplier) that defines the tasks to be performed, the items to be delivered, the acceptance criteria to be applied to delivered items, and other requirements affecting the development or procurement of system products.

assign: Designate a function, product, process, or other item as accountable for a particular purpose.

NOTES

1 The terms *allocate* or *partition* are used in some domains to denote this concept.

2 The "assign" relationship can be in various forms: a) requirement to function, b) requirement to product or process, c) requirement to interface, d) function to product or process, e) function to external entity (e.g., the operator), or f) requirement to external entity (e.g., external system).

associated processes: Processes that enable one or more end products to be put into service, maintained in service, or retired from service.

building block: A representation of the conceptual framework of a system that is used for organizing the requirements, work, and other information associated with the engineering of a system. An element in the structured decomposition of the system.

configuration management: A management process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational management information throughout its life. Reference: ANSI/EIA-649.

constraint: (1) A restriction, limit, or regulation imposed on a product, project, or process.
(2) A type of requirement or design feature that cannot be traded off.

customer: An individual, organization, or enterprise that (1) commissions the engineering of a system; (2) is a prospective purchaser of the end products of a system, or portions thereof; or (3) is an acquirer of a product.

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

derivative system: A special type of precedented system derived from a previously operational system through the use of major elements, but whose requirements have been modified to meet new objectives.

derived requirement: (1) A requirement that is further refined from a primary source requirement or a higher-level derived requirement.
(2) A requirement that results from a design decision for a logical or physical solution representation.

developer: An enterprise or organization that performs the process requirements of this Standard.

development: The action by which a set of requirements is translated into a solution definition for a set of products that satisfy stakeholders.

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

NOTE—Documentation is an instance of a document or a collection of documents.

effectiveness analysis: An assessment of how well a product associated with an alternative logical, physical, or design solution is expected to perform or operate, given an anticipated usage scenario.

enabling product: Item that provides the means for a) getting an end product into service, b) keeping it in service, or c) ending its service.

NOTE—Enabling products are related to the associated processes: development, production, test, deployment, training, support, and disposal.

end product: The portion of a system that performs the operational functions and is delivered to an acquirer.

end product validation: Confirmation by examination and provision of objective evidence that the specific intended use of an end product (developed or purchased), or an aggregation of end products, is accomplished in an intended usage environment.

NOTES

1 The key difference between end product validation and end product verification is that end product validation answers the question: Does the delivered end product conform to the validated input acquirer requirements, certification criteria, or acceptance criteria, as applicable? End product verification answers the question: Does the output end product comply to the output specified requirements from which the end products were built, coded, procured, or assembled and integrated?

2 End product validation is used to demonstrate that the product developed or purchased satisfies the validated acquirer requirements in the context of its intended use.

3 Validation against other stakeholder requirements, generally, is not required. These requirements generally act as constraints on either the solution or the process by which a solution is generated. Constraints on solutions will show up in specifications to which an end product is built, coded, or assembled, and then verified against. Process constraints will be evaluated during management reviews or in management reports.

4 Validated is used to designate the corresponding status.

end product verification: Confirmation by examination and provision of objective evidence that the specified requirements to which an end product is built, coded, or assembled have been fulfilled.

NOTES

1 End product verification is used to demonstrate that the specified requirements (specifications) generated by the developer and used to build, code, or assemble the end product have been satisfied.

2 Verified is used to designate the corresponding status.

engineering life cycle: A sequence of phases that evolves an instance of a system from a concept to a set of products consistent with the exit criteria established for an enterprise-based life cycle phase.

engineering plan: The plan for implementing the processes for engineering a system. The engineering plan reflects an integrated technical effort that balances all factors associated with meeting life cycle requirements.

enterprise: The entity that has governance over a set projects, or over organizations in which projects are carried out.

enterprise-based life cycle: The incremental progress of a system from conception through disposal, marked by management-established milestones with assigned exit criteria.

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

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

functional requirement: A requirement that defines what system products must do and their desired behavior in terms of an effect produced, or an action or service to be performed.

NOTES

1 An example of a behavior is “system switches from standby mode to run mode;” an example of an effect produced is “cause an alert signal;” an example of an action or service to be performed is “signal opens valve.”

2 A functional requirement can include the actor that is to perform the function, the function to be performed, and, if appropriate, the object acted upon. In addition, this information can be complemented by a statement of the environment within which the function is performed, the conditions that cause the function to start, the performance requirements associated with that function, and the conditions that cause the function to terminate.

information database: A repository that provides a capacity to maintain work products and outcomes from implementation of the processes for engineering a system in a controlled manner.

NOTE—This database provides the basis for controlled maintenance of the information needed by the multidisciplinary teams and management to efficiently and effectively accomplish their assigned tasks. It typically contains the requirements, configurations of a system (past, current, and planned), and all analyses and test results. This database allows for traceability, supports the validation and verification tasks, is essential for change management, and provides information to support decision making.

interface requirement: A requirement that defines the conditions of interaction between items.

NOTES

1 Interface requirements include both logical and physical interfaces. They include, as necessary, physical measurements, definitions of sequences of energy or information transfer, and all other significant interactions between items.

2 There are interfaces between a system and things external to the system, and between elements within a system. The latter include, but are not limited to, interfaces between the end products and their operators or maintainers, the interfaces between items that make up an end product, and interfaces between an end product and enabling products of the associated processes.

3 For example, communications interfaces involve the movement and transfer of data and information within the system, and between the system and its environment. Proper evaluation of communications requirements involves definition of both the structural components of communications (e.g., bandwidth, data rate, distribution, etc.) and content requirements (what data/information is being communicated, why it is being moved among the system components, and the criticality of this information to system functionality).

layer of development: (1) A level of abstraction as it relates to the system structure made up of building blocks.
(2) A level of system decomposition.

method: Techniques that support implementation of process tasks.

NOTE— A method is the "how" of each task. Methods have the following attributes: a) thought patterns or approaches; b) knowledge base; c) rules and heuristics; d) structure and order; and e) notation.

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

normative: That portion of a standard or specification that governs implementation.

NOTE—A standards document usually contains three kinds of material: (1) The standard itself (normative part); (2) explanatory material to help the user understand the standard (informative part); and (3) other material concerning the administration of the standard and the sponsoring organization (administrative part). The explanatory material is contained in Notes or "informative annexes." Conformance to a standard is judged solely on the basis of the normative material in the standards document.

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

performance requirement: A requirement that defines how well the system products are required to perform a function, along with the conditions under which the function is performed.

precedented: An end product that is a legacy product undergoing modification or a product that the enterprise both has the expertise to make and has similar products already in the market place.

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

product: (1) An item that consists of one or more of the following: hardware, software, firmware, facilities, data, materials, personnel, services, techniques, and processes.
(2) A constituent part of a system.

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

NOTE— For the purposes of this Standard, project and program are synonymous.

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

requirement: (1) Something that governs what, how well, and under what conditions a product will achieve a given purpose.
(2) Normative elements that govern implementation of this Standard, including certain documents such as agreements, plans, or specifications.

requirements validation: Confirmation by examination that requirements (individually and as a set) are well formulated and are usable for intended use.

NOTES

1 See Table C.25 for what constitutes "well formulated."

2 There are five types of requirements validation in this Standard stated in Requirements 25 through 29.

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

risk aversion: The act of averting risk. Averting risk can be through various means: mitigation, avoidance, transfer, or acceptance.

risk management: An organized process for identifying and assessing risks, and for implementing means to avoid them or mitigate their effect if they occur.

specification: A document that contains specified requirements for a product and the means to be used to determine that the product satisfies these requirements.

stakeholder: An enterprise, organization, or individual having an interest or a stake in the outcome of the engineering of a system.

NOTES

1 Examples of stakeholders are acquirer, user, customer, manufacturer, installer, tester, maintainer, executive manager, project manager, and all other personnel having a stake in the development or outcome of the engineering of a system. The enterprise as a corporation or agency and the general public are also stakeholders.

2 An acquirer (see definition above) is a specific instance of a stakeholder and is individually acknowledged since the acquirer is a principal in establishing the acquirer-supplier agreement.

3 All stakeholders other than the acquirer are referred to as "other stakeholders".

stakeholder requirement: A requirement that represents what stakeholders of a system need or expect of the system products.

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

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

supplier: Provides a product (either end products, enabling products, or both) or a group of products to an acquirer. The supplier (external or internal to the acquirer's organization) can be a vendor that has a product that does not need development, or a developer that must develop the desired system product or products.

system: An aggregation of end products and enabling products to achieve a given purpose.

system technical requirement: A requirement derived from one or more stakeholder requirements and stated in technical terms.

technical performance measurement (TPM): The technique of predicting the future value of a key technical parameter of the higher-level end product under development, based on current assessments of products lower in the system structure.

NOTES

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

2 Key characteristics of TPM are:

a) *Achievement to Date*—present achieved value of the technical parameter based on estimates or actual measurement;

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

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

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

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

f) *Objective*—the goal or desired value at the end of the technical effort;

g) *Threshold*—the limiting acceptable value that, if not met, would jeopardize the project;

h) *Variation*—the difference between the planned value and the achievement-to-date value.

technical review: An event at which the progress of the technical effort is assessed relative to its governing plans and technical requirements.

test article: An item built, constructed, coded, or otherwise implemented, for checking conformance to specified requirements or for checking validation against acquirer requirements for the item.

traceability: The ability to identify the relationship between various artifacts of the development process, i.e., the lineage of requirements, the relationship between a design decision and the affected requirements and design features, the assignment of requirements to design features, the relationship of test results to the original source of requirements.

unprecedented: A specific end product that is not known *a priori*, or the enterprise has limited experience in developing this type of system.

user: Individual, organization, or enterprise that uses, applies, or operates system products.

validation: See *end product validation* and *requirements validation*.

verification: See *end product verification*.

Annex B—Enterprise-based life cycle (informative)

The various commercial and non-commercial enterprises, within widely diverse domains, have similar enterprise-based life cycles, and generally exist for the same purpose. That purpose is to incrementally develop systems and control passage from one increment to another so as to reduce risk, control costs, and provide and maintain system products that will be competitive and provide user satisfaction throughout the life cycle.

Each enterprise-based life cycle is characterized by distinct phases marked by established exit criteria and management reviews to ensure that the exit criteria are satisfied prior to making a decision on whether or not to approve progress to the next phase or sequence of phases, or to make modifications or improvements to maintain competitiveness. Although the various enterprise-based life cycles might have different named phases, and different phase and life cycle time periods, most, if not all, have these five distinct functional phases: (1) assessment of opportunities, (2) investment decision, (3) system concept development, (4) subsystem design and pre-deployment, and (5) deployment/installation, operations, support, and disposal.

B.1 Relationship to engineering life cycle phases

Figure B.1 shows five typical phases of an enterprise-based life cycle and the use of appropriate engineering life cycle activities to meet the exit criteria for the enterprise-based life cycle phases.

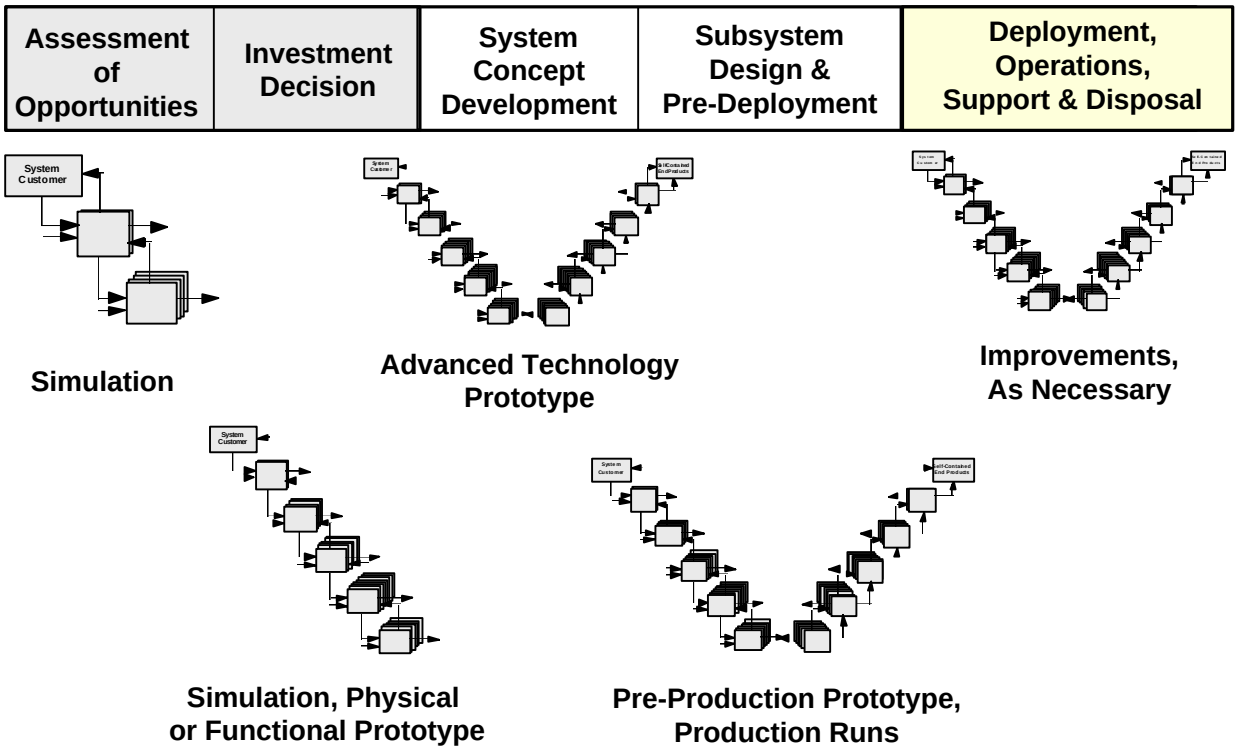

Figure B.1—Enterprise-based life cycle phases

NOTES

1 It is during the investment decision phase that a commercial or subsystem supplier organization typically prepares a proposal in response to a competitive solicitation, when such governs an enterprise's activities. In other organizations, such as government agencies, solicitation and proposal activities can occur before any of the above phases when competition is deemed appropriate.

2 Enterprise-based life cycles tend to be unique to an enterprise, and are subdivided by different phases that depend on the needs of the enterprise. These are generally based on the enterprise's own external environment. They are established, for example, in response to market cycles, government agency directives, or fiscal considerations. They are not generally based on engineering efforts required for a system development (or portions thereof), but on entry and exit criteria to meet internal or customer-driven milestones.

3 The key message of Figure B.1 is that appropriate engineering life cycle process activities are completed to meet the exit criteria of each enterprise-based life cycle phase, regardless of the name or purpose of the phase.

B.2 Product evolution

During the early phases of this generic life cycle, various levels of system products are developed. For instance, during the first phase (assessment of opportunities), a simulation-produced prototype can be used to identify, qualify, and select new or improved system and business opportunities.

During the second phase (investment decision), a physical or functional prototype can be developed to understand a solution so that determination can be made whether to continue with the development and so that project plans are produced in preparation for transition to system development. For competitive developments, a bid or no bid can be determined and a proposal can be developed, if necessary.

During the third phase (system concept development), an advanced technology prototype can be developed, including one sufficiently operational to assess performance and cost factors and to identify and reduce critical risk factors.

The fourth phase (subsystem design and pre-deployment) produces a pre-production prototype, which will be used for verifications and validations and acceptance by the acquirer, and required production volume of end products and enabling products for deployment or installation.

The last phase (deployment/installation, operations, support, and disposal) is where the system products are deployed or installed for operation and various operational, maintenance, and disposal support provided, as required. During this last phase, reengineering is often necessary to keep the products competitive and useful. If needed, the processes of this Standard are applied while using the appropriate engineering life cycle phases.

B.3 Life cycle considerations

Cost is an important criterion for both making a decision to develop a certain system and for buying that system. Various heuristics attribute that from 60 to 80 percent of the life cycle cost of a system is experienced in the operations and support phase. It is essential, therefore, that focus be on ways to reduce such costs during the earlier phases of the enterprise-based life cycle. It is important to treat cost, especially pertaining to associated processes, as an independent variable while making tradeoff analyses.

Annex C—Process task outcomes (informative)

This Annex provides an informative set of representative tasks and their expected outcomes for the thirty-three requirements of Clause 4.

C.1 Acquisition and Supply task outcomes

Table C.1—Requirement 1 (Supply Process - Product Supply)

Representative tasks	Expected outcomes
a) Assess acquisition request, offer, or directive	The capability of the enterprise, organization, project, or team to provide a system, or portion thereof, that meets acquisition document requirements within the stated constraints and the enterprise strategic plan and business strategy, or within the project plan and constraints, or within the team charter, as applicable, is determined. Includes, as appropriate: 1) engineering and other applicable technical and project plans that allow determination of engineering and management tasks, costs and schedules, resource requirements, and technical capabilities and capacities (invoke applicable Planning Process tasks); 2) decision whether to work with the acquirer to provide the desired system, or a portion thereof, based on established enterprise criteria or on project or team capability; 3) resolution of added or changed requirements and areas of concern. 4) preparation and submission of an appropriate technical and cost response in accordance with acquisition request, enterprise business strategy, and enterprise policies and procedures, or with project plans, policies, and directives.
b) Negotiate agreement	A satisfactory agreement is established based on the bounds determined by, as applicable: 1) applicable legal, regulatory, policies, procedures, and practices that will affect negotiation strategy or conduct; 2) the type of agreement to be negotiated; 3) negotiation strategy; 4) conditions identified from the plans for the procurement work effort that could affect negotiations and agreement performance; 5) constraints identified from the plans for the procurement work effort that could affect negotiations and agreement performance.
c) Record agreement	Established agreement is captured in a form and medium appropriate to the effort.
d) Implement agreement	A project established and processes (including replanning, as necessary) activated to complete the requirements of the agreement.
e) Deliver products and other deliverables per agreement	Agreement requirements satisfied by delivery of required products and other deliverables in accordance with agreement instructions.

Table C.2—Requirement 2 (Acquisition Process - Product Acquisition)

Representative tasks	Expected outcomes
a) Prepare acquisition requests, offers, or directives	Acquisition documents, as applicable to the technical effort, prepared to include: 1) plans to be provided to suppliers, as applicable; 2) purpose of the acquisition, the essential requirements to be met, the products to be delivered by a supplier, and the operational concept and expected operational environment for each product, as applicable; 3) what the products to be delivered must be able to do; how well the products must perform; desired characteristics of the products, constraints, and other essential product attributes; management concerns including line of authority, financial management, and reporting; and requirements that can affect the cost, schedule, and risk in accomplishing the work effort or delivery of the product; 4) concerns such as cost and schedule that can constrain the work effort or product, and states whether or not the concern can be traded off; 5) expected tasks or work to be done by the supplier; 6) the data and other work products to be delivered, including form, format, and schedule.
b) Evaluate supplier response	Supplier or suppliers selected that will do the agreed-to work and provide the desired products, as appropriate.
c) Make offer or provide directive	Offer made or directive provided to the selected supplier or suppliers.
d) Negotiate agreement	A satisfactory agreement established based on the bounds determined by, as appropriate: 1) applicable legal, regulatory, policies, procedures and practices that will affect negotiation strategy or conduct; 2) the type of agreement to be negotiated; 3) negotiation strategy; 4) conditions identified from the plans for the procurement work effort that could affect negotiations and agreement performance; 5) constraints identified from the plans for the procurement work effort that could affect negotiations and agreement performance.
e) Record agreement	Established agreement is captured in a form and medium appropriate to the effort.
f) Accept delivered products	Installed or delivered system products validated as satisfying user, customer, or assigned

Table C.3—Requirement 3 (Acquisition Process - Supplier Performance)

Representative tasks	Expected outcomes
a) Define supplier relationships	The type of supplier support required, level of participation, procedures and criteria for selection and control, procedures for participation, as appropriate, on developer's multidisciplinary teams, and an appropriate acquirer-supplier agreement are established.
b) Participate on product teams	Agreed-to procedures for participation of supplier personnel on developer multidisciplinary product teams and for participation of developer personnel on supplier multidisciplinary product teams are implemented.
c) Monitor product metric data	Supplier performance against product metrics established in the agreement is determined. Invoked the applicable tasks in the Assessment Process.
d) Flow-down changes in requirements of operational concept	Assurance made that all requirement and operational concept changes affecting the supplier's project have been properly communicated to the supplier.
e) Control requirement changes	All changes approved to functional and performance requirements and to constraints, made by the supplier, that would affect the developer's project or other related projects or products. Approved changes have been appropriately distributed and implemented.
f) Assess progress against requirements	Progress against assigned requirements included in the agreement and as changed by established change procedures is determined. Required technical reviews completed. Invoked applicable tasks of the Assessment Process.
g) Validate products received	Assurance made that delivered products satisfy assigned requirements and approved changes. Resolution of identified variations resulting from validation of the delivered product is completed. Invoked the applicable tasks of the End Products Validation Process.

C.2 Technical Management task outcomes

Table C.4—Requirement 4 (Planning Process - Process Implementation Strategy)

Representative tasks	Expected outcomes
a) Identify stakeholders	Intended users or customers and other stakeholders who will have an interest or stake in the outcome of the project are established.
b) Identify applicable documents	Applicable source and technical documents and the requirements therein that could affect the project effort are identified and acquired, including: 1) the scope and purpose of both the project and products to be developed or reengineered 2) stated purpose of the products, expectations of the stakeholders, expected benefits to stakeholders, as well as the goals and objectives of the system, or portion thereof, to be developed or reengineered 3) enterprise policies, priorities, and constraints on funding, personnel, facilities, manufacturing capability and capacity, and critical resources that will affect accomplishing the requirements and goals of the source and technical documents 4) (a) applicable processes, standards, and specifications; (b) core enterprise technologies; (c) risks to business growth by new project; (d) must-win criteria; (e) net cost targets; (f) methods of resource allocation; (g) how work and changes will be authorized; (h) how information will be captured; (i) how work packages will be formed and controlled; (j) scope and procedures for tradeoff analyses, effectiveness analyses, and risk management, based on enterprise goals and planning baselines.
c) Identify associated process approaches	How development of enabling products associated with production, test, deployment/installation, and logistics processes will be implemented is determined.
d) Identify applicable life cycle phases	Applicable enterprise-based life cycle phases (see Annex B.2), the expected work product outputs and management reviews, and the relevant exit criteria for each applicable enterprise-based life cycle phase, including level of product maturity expected, level of acceptable risk, management review concerns, and documentation requirements, are determined.
e) Identify and define technical process and project integration	How the applicable processes of this Standard will be integrated with each other and with other processes specified in enterprise and agreement documents, and which internal and external projects that will be involved and how they will be integrated are determined.
f) Identify and define progress assessment	Required reporting requirements, specific product and process metrics to be used, how and when metrics will be collected and by whom, and how progress will be assessed are determined.
g) Prepare the process implementation strategy	A process implementation strategy document based on the integrated results of the outcomes of the above tasks is prepared.

Table C.5—Requirement 5 (Planning Process - Technical Effort Definition)

Representative tasks	Expected outcomes
a) Identify project requirements	The following are determined: 1) specific requirements include (a) work that the supplier is required to accomplish, (b) functions of the system, or portion thereof, to be furnished, engineered, or improved; how well the functions are to be performed; any required physical characteristics; and the operations concept, (c) data to be delivered and when, (d) budget and schedule requirements, and (e) other technical requirements provided in acquirer-supplied planning documents; 2) other stakeholders who have or who will have requirements or expectations with respect to the work to be accomplished or the system to be provided (for example, local, national, or international government agencies; persons living or working in the areas near where system products will be used or where products will be developed and produced; commercial or military competitors; and employees involved with the project); 3) potential conflicts between the acquirer-supplier agreement (proposed or final), the process implementation strategy, and enterprise policies and procedures, core technologies, and capacities; 4) specific constraints and any conflict between the process implementation strategy and the agreement (proposed or final) with respect to development, production, test, deployment, support, or disposal of the system products to be delivered, or the training of personnel required to operate and maintain the products.
b) Establish information database	The types and quantity of data and schema and other information that will have to be recorded and maintained are determined; a database that can securely retain and make available project information, as required, is established.
c) Define risk management strategy	The following are determined: (1) how the technical risk areas of the technical effort will be identified and tracked and (2) the appropriate risk aversion approaches based on the acceptable levels of risk specified in the agreement or in enterprise policies and procedures.
d) Define product and process metrics	The following are defined: (1) product metrics by which the quality of the products is to be evaluated; (2) process metrics by which the efficiency and effectiveness of the tasks of the technical effort are to be evaluated; and (3) frequency and methods by which product and process metrics are to be collected.
e) Establish cost objectives	Rigorous cost goals (ownership, acquisition, operating, support, and disposal) to be used in tradeoff analyses are established.
f) Identify technical performance measures	The following are determined (1) technical objectives related to success of the system, or portion thereof, [e.g., measures of effectiveness (MOEs) by which the user, customer, or acquirer will measure satisfaction or acceptance]; and (2) critical performance parameters that will receive management focus and are to be tracked using Technical performance

g) Identify applicable tasks	The following are identified: (1) key events of the project (e.g., technical reviews, physical integration, major test, product and process verifications, and end product validations) established by input planning documents; (2) entry and exit completion criteria for each event; and (3) tasks required for meeting the entry and exit criteria of each event and for accomplishing each applicable process. NOTE—The following structure of tasks can be helpful for accomplishing scheduling, staffing determination, and resources required: 1 Key events required to meet technical requirements (e.g., test and technical review) 2 Primary tasks related to accomplishing entry and exit criteria of each key event (e.g., define stakeholder requirements and prepare engineering drawings) 3 Support tasks that enable the staff accomplishing primary tasks to meet their objectives (e.g., provide resources, equipment, facilities; acquire appropriately skilled personnel for accomplishing the primary tasks; and arrange travel). 4 Management tasks required to direct, monitor, review, and approve the primary and support tasks (e.g., serve as chair of a technical review, and review and approve documents for transmittal to the customer).
h) Identify methods and tools	The following are determined: (1) appropriate methods for accomplishing identified tasks, or groups of tasks of each applicable process; (2) required automated tools; (3) required specialized facilities and equipment; and (4) training requirements.
i) Establish technology insertion approaches	The applicable or potential technology constraints are identified and the approach for conducting parallel technology developments, and planned technology insertions are established.

Table C.6—Requirement 6 (Planning Process - Schedule and Organization)

Representative tasks	Expected outcomes
a) Develop event-based schedule	The key events for the technical effort associated with applicable enterprise-based life cycle phases, related applicable tasks to each event, and the completion criteria for each task and an event are developed and documented.
b) Develop calendar-based schedule	The calendar date that each key event will be completed or expected to be completed; the planned start and completion time for accomplishment of each task (primary, support, and management); and the dependency relationships between tasks and between tasks and events, and between events with other events are developed and documented.
c) Identify resource requirements	The material resources, facilities, and equipment required to complete each scheduled primary, support, and management task are determined, and the date such resources are required is specified.
d) Define staffing and discipline needs	The following are determined: (1) personnel needs by discipline and performance level to complete scheduled primary, support, and management tasks, and the date each staffing need is required; (2) internal and external supplier training needs and schedules to achieve required proficiencies; and (3) risk to the project, if adequate staffing is not available.
e) Define team and organization structure	(1) The multidisciplinary teams needed to carry out the planned technical efforts and produce required work products are formed within enterprise and project resource constraints; (2) The composition of teams by functional and disciplinary membership that are organized to support specific system product development is established; (3) The names of staff members assigned to each team are established; (4) Responsibilities and authority of teams and team members are defined; and (5) Roles, responsibilities, authority and boundaries for each team are established.

Table C.7—Requirement 7 (Planning Process - Technical Plans)

Representative tasks	Expected outcomes
a) Develop Engineering Plan	An efficient and economical means of implementing the processes for engineering a system is defined and documented. It answers the following questions: 1) What is the general problem to be solved? 2) What is the benefit to the acquirer (enterprise perspective)? 3) What is the application context of the general problem to be solved? 4) What is the boundary of the general problem to be solved, denoting what can be controlled by the developer (inside) and what influences the development and is influenced by the development but not controlled by the developer (outside)? 5) What are the required inputs and outputs? 6) What are the influencing factors and constraints? 7) How are the system concerns, as appropriate, of reliability, availability, maintainability, security, safety, health factors, survivability, electro-magnetic compatibility, radio frequency management, and human factors being considered and included? 8) What processes and tasks must be accomplished? 9) How will each process be accomplished? 10) What resources, methods, and tools are necessary to accomplish the tasks of each process? 11) How will the required resources and tools be acquired? 12) What is the organizing structure? 13) How will the organization be staffed and managed? 14) What are key intermediate events leading to project completion, and how will their occurrence be determined? 15) When, where, and by whom will tasks and events be completed? 16) What are the risks involved? How will risks be managed? 17) What are the completion criteria for the process tasks? 18) What are the entry and exit criteria for reaccomplishing each process? 19) How will project completion be determined? NOTES 1 The engineering plan usually covers one or more phases of the enterprise-based life cycle and the applicable phases of the engineering life cycle. 2 The engineering plan is to cover process applications within the engineering life cycle to meet the exit criteria of the applicable enterprise-based life cycle phases, as consistent with the acquirer-supplier agreement and the extent of the project conducted within an enterprise.
b) Develop Risk Management Plan	Documentation of the tasks to be accomplished by project teams and analysts for identification of potential risks, characterization and prioritization of identified risks, aversion of risks, tracking and control of risks, and communication of risk status are defined and documented.
c) Develop Technical Review Plan	The tasks to be accomplished to implement required technical reviews and a detailed description for each review are developed and documented to include: (1) a check list for tasks to be accomplished, (2) entrance and exit criteria, (3) review schedule, (4) documentation requirements, (5) distribution list for technical data package, (6) participants, and (7) responsibilities of participants.
d) Develop Validation Plans	The tasks to be accomplished and the resources to be allocated and scheduled for validating that: (1) the system technical requirements, logical representations, and derived technical requirements are well formulated (see Requirement 25) and conform to their respective sources, and (2) the products received from suppliers, or delivered to an acquirer, conform to the user, customer, or assigned requirements associated with the end product are defined and documented.

e) Develop Verification Plans	The tasks to be accomplished and the resources to be allocated and scheduled for verifying that (1) the selected and characterized physical solution description satisfies the assigned system technical requirements, logical representations, and derived technical requirements, (2) end products satisfy their specified requirements, and (3) enabling products will be ready when required to provide life cycle support to their respective end products are defined and documented.
f) Develop Other Applicable Plans	The tasks to be accomplished to complete required control activities or other design activities such as design-to-cost, Technical Performance Measurement, technology insertion, safety, security, human factors engineering, and maintenance reliability (see Annex D for others), as required in an agreement or by enterprise policies and procedures, are defined and documented.

Table C.8—Requirement 8 (Planning Process - Work Directives)

Representative tasks	Expected outcomes
a) Develop work packages	The work required, input sources, schedules, budget, and reporting requirements to implement, execute, and control the work are defined and documented.
b) Generate work authorizations	Approval/disapproval of work packages is assigned, and work authorizations are documented.

Table C.9—Requirement 9 (Assessment Process - Progress Against Plans and Schedules)

Representative tasks	Expected outcomes
a) Identify events, tasks, and process metrics for monitoring	The events and tasks that must be monitored, as well as the metrics that will be used to assess progress against plans and schedules, are identified. The applicable expected values for each progress metric are established.
b) Collect and analyze process metrics data	Results from completion of required tasks and events, and process metrics data are determined and tracked.
c) Compare process metrics data against plans and schedules	The following are determined: (1) completion of required tasks and events, (2) variances of metrics from expected values, (3) progress variances from plans and schedules, (4) technical areas requiring management or team attention, and (5) cost and schedule risks.
d) Implement required changes	The cost effective changes to correct variances and needed changes to plans and schedules, and required changes, revised work directives, and updated plans to reflect approved changes and management decisions are identified, approved, and implemented.

Table C.10—Requirement 10 (Assessment Process - Progress Against Requirements)

Representative tasks	Expected outcomes
a) Identify product metrics to be monitored	Product-related metrics, and their expected values, that will affect the quality of the product and provide information of the progress toward satisfying user/assigned requirements, other stakeholder requirements, and derived requirements are identified and documented.
b) Collect and analyze product metrics data	The following are determined, as appropriate: (1) analyzed, estimated, or measured values of critical performance parameters at predetermined events (e.g., simulation and prototype tests), (2) compliance to applicable requirements, (3) levels of technical risks, (4) marginal cost benefit of performance beyond requirements, (5) degree of customer satisfaction and public acceptance, and (6) effect of a critical performance parameter status on related end-user products.
c) Record rationale for decisions and assumptions made	The following are recorded, as applicable: (1) rationale for selection of alternatives based on recommendations and effects of tradeoff and effectiveness analyses and (2) assumptions associated with decisions made during requirements definition, solution definition, tradeoff analyses, effectiveness analyses, verifications, and validations.
d) Compare results against requirements	The following are determined, as applicable: (1) satisfaction of technical requirements, (2) progressive maturity of the system, or portion thereof, being engineered/reengineered, (3) variances from expected values from Technical Performance Measurements, and (4) variations from requirements resulting from end product verifications and end product validations.
e) Identification and Implementation of Required Changes	The following are identified, evaluated, and implemented, as applicable: (1) alternative corrective actions to mitigate out-of-tolerance Technical Performance Measurements, (2) other changes to be implemented so that products will meet requirements, (3) recommended user/assigned, other stakeholder, or technical requirement changes, and (4) implementation of revised specifications and configuration baselines that reflect approved changes and management decisions.

Table C.11—Requirement 11 (Assessment Process - Technical Reviews)

Representative tasks	Expected outcomes
a) Identify technical review objectives and requirements	The following are identified and documented: (1) purpose and objectives of the review, (2) agenda requirements, (3) tasks to be completed at each required review, (4) entrance and exit requirements, (5) documentation requirements, (6) distribution requirements, and (7) responsibilities of review participants.
b) Determine progress against event-based plan	The satisfaction of entrance requirements to the review are determined and documented.
c) Establish technical review board, agenda and speakers	For each review, the following are established: (1) persons who will participate in the review, (2) chairperson, (3) secretary, (4) reviewers of the presentation, (5) agenda that meets review requirements and ensures that all required tasks are completed, and (6) members of the design team that will prepare the data package, prepare the presentation, prepare material for distribution at the review, make the presentation, answer questions, and accomplish tasks to close out action items**.**
d) Prepare technical review package and presentation material	Comprehensive read-ahead material is prepared that includes sufficient information so that technical board members can understand the design and participate effectively in the review. Review team responsibilities, agendas, plans, and expectations from the review; are defined and documented. A comprehensive set of presentation materials that describe the assigned design topics and that satisfy review objectives is prepared.

e) Facilitate resolution of emerging issues	Emerging issues identified and resolved prior to the review.
f) Conduct technical review	The following are assessed by the review: (1) maturity of system, or portion thereof, being engineered, (2) progress according to plans and requirements, (3) risks and variances in cost, schedule, and performance, and (4) readiness to proceed with the next phase of development. Action items required to meet review objectives are generated, recorded, and assigned.
g) Close-out review	The following are completed for review close-out: (1) preparation and distribution of minutes that include purpose, time, place, attendees, decisions, action items, due date, and persons responsible for resolving actions items, (2) resolution of action items, and (3) sign off by chairperson.

Table C.12—Requirement 12 (Control Process - Outcomes Management)

Representative tasks	Expected outcomes
a) Capture process outcomes	The following are recorded in the information database: (1) the outputs of the technical processes implemented in the engineering of a system, (2) the methods, tools, models, and metrics used, (3) recommendations, decisions, assumptions, and effects, (4) lessons learned, and (5) other data that allows traceability of requirements.
b) Perform configuration management	The configuration of the products is documented and made available. The following is realized: (1) product configuration is known and reflected in product information; (2) beneficial product changes are effected without adverse consequences; (3) change is managed from the first implemented phase during system design, (4) information that will be needed to make later decisions on products is captured; (5) consistency between a product and information about the product; and (6) capability to distinguish between product versions or builds. NOTE—ANSI/EIA-649 can be used in conjunction with this Standard, for configuration management.
c) Perform change management	Traceability of changes is maintained and controlled, including source of the change, processing methods, approvals, and implementations in accordance with the Change Management Plan.
d) Perform interface management	System internal and external interfaces are maintained and controlled, including completion of interface definition, assessments of compatibility, changes, and coordinations and approvals in accordance with the Interface Management Plan. Interfaces are managed, ensuring that: (1) all internal and external functional and physical (including human) interfaces for a building block are identified, defined, assigned, documented, and managed; (2) building block design definitions are compatible in terms of form, fit, and function; and (3) interface changes affecting the building block and affected by the building block (see Clause 6) are controlled to prevent adverse consequences.

e) Perform risk management	Potential risks are identified, characterized and prioritized, and properly averted, tracked and controlled. Risk status is communicated in progress reports, in proposals, and at technical reviews, in accordance with the Risk Management Plan. A clear view of future risks is provided, better decision making is enabled, and problems are prevented from occurring—but if they do occur, a plan exists to mitigate the effect of the problem. NOTES 1 Risk is always present in an engineering or reengineering project. Sources of risk include the tendency of acquirers to: (1) desire products of a system that are intended for technical accomplishment near the limits of the state of the art (performance); (2) push for delivery of system products as soon as possible to meet an imminent market opportunity or threat; and (3) limit funding available. Additionally, risks come from both internally and externally imposed constraints (e.g., resource, capacities, environmental conditions, and reuse). 2 The major sources of risk are programmatic, schedule, political, financial, and technical. Risks are greater when planning, control, resources, and time are inadequate. Risks are also greater when information is not available for decision making, or when the information is too much, too little, too late, irrelevant, or inaccurate.
f) Perform data and document management	Data and documents are maintained and controlled, including development support, handling and storage, and required technical data and document delivery in accordance with the Data Management Plan. Data and document management includes capturing data and documents generated during implementation of the processes of this Standard, and generating and maintaining an evolving technical data package. A typical technical data package includes: (1) a buy-to description (e.g., detail specifications and/or final drawings); (2) a build-to description; (3) design documentation; (4) engineering changes, deviations, and waivers; and (5) enabling product descriptions. Build-to descriptions include: (1) models, drawings, and specifications; (2) production planning; (3) tool design; (4) bill of materials; and (5) statistical process control plan. NOTE—Multidisciplinary teamwork is essential to ensure the accuracy and completeness of technical manuals and the technical data package.

g) Manage information database	Relevant data and information are maintained and controlled for the project, including inputs and outputs of control process tasks and ensuring back-ups, if applicable, of digital databases. Relevant data includes: 1. Inputs and outputs of technical process activities: a) work products (e.g., specifications, drawings, and code lists); b) archival data (e.g., decisions made [including rationale], assumptions, lessons learned, changes, and empirical data); c) stakeholder requirements (e.g., technical objectives, constraints, and interfaces); d) requirement, functional, and physical architectures; e) physical models developed (e.g., prototypes, breadboards, brassboards, and mock ups); f) simulation model outputs and assumptions; g) metrics (e.g., cost and technical performance measures); h) planning documents (e.g., engineering plan and technical event plan); i) technologies. 2. Process models used for: a) analysis of problem (analysis of requirements and analysis of functions) (e.g., Quality Function Deployment, behavior, and time); b) solution definition (synthesis) (e.g., for design); c) validation and verification; d) systems analysis (e.g., for tradeoff analyses, risk analyses, and effectiveness analyses); e) control (e.g., interfaces, data, configurations, schedules, costs, product performance, reviews, and assessments). 3. Tools used: a) automated tools (e.g., traceability, analysis, and design); b) validation and verification tools; c) tradeoff analysis support tools; d) communication tools; e) status reporting/projection tools.
h) Manage and track requirements	The following are maintained and controlled: (1) input requirements (acquirer and other stakeholder), system technical requirements, logical solution representations, physical logical solution representations, derived technical requirements, and specified requirements; (2) validation results; (3) requirement changes resulting from resolution of variances; and (4) changes made to requirements through formal change procedures from Configuration Management, Change Management, and Interface Management tasks.

Table C.13—Requirement 13 (Control Process - Information Dissemination)

Representative tasks	Expected outcomes
a) Provide progress status	Process and product metric data have been disseminated according to the agreement, engineering plan, and enterprise policies and procedures, and to meet approved requests.
b) Provide planning information	Work packages and appropriate technical plans have been disseminated to project teams and other required or approved recipients.
c) Disseminate approved and controlled requirements	Acquirer/assigned, other stakeholder, system technical and derived technical requirements, and all changes to requirements are distributed in a timely manner to all stakeholders to ensure that all work is conducted in accordance with the latest approved requirements.
d) Provide information for and from reviews	The following have been disseminated, as appropriate: (1) read-ahead technical review package to technical review board members, (2) information and items necessary to demonstrate that event-based criteria have been satisfied for initiation of the review, (3) information packages and presentation materials at the review, (4) minutes of the review, action items required for closure, and final close-out approval.
e) Make available design data and schema	Data pertinent for the technical effort have been disseminated to project teams and team members to ensure information availability for decisions and events, and to other authorized recipients requesting information
f) Make available lessons learned	Lessons learned have been disseminated to other projects within the enterprise and to other teams within the project.
g) Report variances	Product and process variances have been reported along with (1) recommended actions to return the product or process metric to established expectations or requirements, (2) cost and schedule impacts, and (3) effects on the project if no action is taken.
h) Disseminate data deliverables	Data deliverables have been disseminated as required by the agreement, enterprise policies and procedures, the engineering plan, and other technical plans.
i) Disseminate approved changes	Approved requirements and design changes and updated plans have been distributed to approved or required recipients
j) Disseminate directives	Work directives resulting from management decisions have been disseminated to intended recipients that initiate or change work by project teams or support organizations within the enterprise.

C.3 System Design task outcomes

Table C.14—Requirement 14 (Requirements Definition Process - Acquirer Requirements)

Representative tasks	Expected outcomes
a) Identify, collect, and prioritize acquirer's system requirements	User, customer, or assigned requirements for a system, or portion thereof, have been identified and defined in terms of needs, expectations, capabilities, and priorities, or of assigned requirements for a system, or portion thereof, as expressed in specifications. Specifically, the following have been identified, as applicable: 1) concept of operation 2) what the acquirer wants the products of the system to accomplish (functional requirements) 3) how well each function must be accomplished (performance requirements) 4) natural and induced environments in which the system products must operate or be used 5) design constraints such as use of non-developmental or reusable items 6) requirements pertaining to availability, electro-magnetic compatibility, health factors, human factors. interoperability, maintainability, reliability, safety, and security 7) measures of effectiveness (MOEs) that reflect overall expectations against which satisfaction will be determined 8) constraints pertaining to development, production, test, deployment/installation, training, support/maintenance, and disposal.
b) Ensure completeness and consistency of the set of collected acquirer requirements	The collected user, customer, or assigned requirements are validated. Resolution of all conflicts and variances is competed. Invoked the Requirements Validation Process, Requirement 26.
c) Record set of acquirer requirements	Validated set of acquirer requirements is captured in the established information database.

Table C.15—Requirement 15 (Requirements Definition Process - Other Stakeholder Requirements)

Representative tasks	Expected outcomes
a) Identify and collect other stakeholders' end product requirements	Other types of requirements that can constrain the engineering of the system's end products are identified, collected, and defined, such as: 1) project plans 2) team assignments and organization 3) automated tools availability and approval for use 4) required metrics 5) decisions from management or technical reviews 6) enterprise standards, guides, policies, and procedures 7) enterprise technologies 8) enterprise physical and financial resources
b) Identify and collect other stakeholders' enabling product requirements	Enabling product requirements associated with manufacturing/production, test, deployment/installation, training, support, and disposal (including disposal) processes including enterprise capacities (facilities, equipment, tools, and staff) to accomplish these processes are identified, collected, and defined.

c) Identify and collect other stakeholders' external constraints	Other end product and development process constraints from external sources are identified, collected, and defined, such as: 1) national and international standards, laws, and regulations (including environmental protection, hazardous material exclusion list, and waste disposal) 2) technology base 3) industry and international standards and general specifications 4) competitor product capabilities and trends 5) interfaces with other existing or evolving systems and platforms
d) Ensure completeness and consistency of the set of other stakeholders' requirements	The collected set of other stakeholder requirements is validated. Resolution of all conflicts and variances is completed. Invoked the Requirements Validation Process Requirement 27.
e) Record set of other stakeholder requirements	Validated set of other stakeholder requirements is captured in the established information database.

Table C.16—Requirement 16 (Requirements Definition Process - System Technical Requirements)

Representative tasks	Expected outcomes
a) Establish required transformation rules, priorities, inputs, outputs, states, modes, and configurations	Transformation rules, priorities, inputs, outputs, states, modes, and configurations that will influence and affect the other tasks for definition of system technical requirements are identified and defined, as appropriate to each system product.
b) Define operational requirements	The range of anticipated uses of the end products, as identified in the concept of operations or specification, or for potential end products, is defined, including for each operational profile, definition of: 1) the utilization environment and factors, natural or induced, that can affect end product performance 2) the events to which end products must respond 3) the physical and functional interfaces (e.g., mechanical, electrical, thermal, data, and procedural) including physical interactions (e.g., form and fit), system boundaries (what is controlled by the developer) and interactions (e.g., information flows and behaviors) of products or environments within developer control and those systems or environments outside system boundaries 4) what system end products must be able to accomplish (functional requirements) to satisfy acquirer identified requirements. Includes factors such as producibility, testability, transportability, installability, operability, supportability, disposability, reliability, availability, maintainability, security, and safety 5) how often end products will be used, cycle time between uses, and how often each product function will be accomplished.
c) Define Performance Requirements	The following are defined: (1) the performance expectations for each functional requirement (how well the function must be accomplished), (2) the set of measure of performance (MOPs), made up of the functional and performance requirement combinations, associated with each MOE, (3) the critical performance parameters (TPMs) selected from the MOPs that will be key indicators of end product or system performance, and if not met, that will cause the associated MOE to not be satisfied and will put the project in cost, schedule, or performance risk, and (4) functional and performance testability approach for each requirement statement.

d) Analyze acquirer and other stakeholder requirements to: 1) Define human factors effects 2) Establish capacities and timing 3) Define technology constraints 4) Define product design constraints 5) Define enabling product requirements 6) Identify conflicts 7) Determine tradeoff analysis criteria	The following are identified and defined, as applicable: 1) the user or operator roles, as applicable, and the human factors effects (ergonomic limitations, work space, eye movement, access, cultural background, natural and induced environmental constraints, work tasks, and time constraints) associated with functional and performance requirements on potential users, operators, installers, or recipients and handlers of system end products 2) required capacities (e.g., memory, storage, and flows) of end products and timing of events, states, modes, and functions related to each operational profile 3) any constraints or limitations from use of existing technologies and the risks associated with using any unproven technologies 4) any constraints that will influence or affect end product design (e.g., materials, special skills, and automated tools), required physical characteristics (e.g., size, color, texture, weight, and buoyancy), operator safety, system security, reuse requirements, standardization of end products, open system architecture, maintainer access, handling and storage, transportability, and other attributes of end products or design processes for which tradeoffs cannot be made 5) technical requirements for enabling products associated with processes to develop, produce, test, deploy/install, operate, support/maintain, train, and retire/dispose of end products under development or being improved 6) conflicts among the requirements set 7) the set of risk, cost, schedule, and performance criteria to be used in conducting tradeoff analyses for conflict resolution. NOTES 1 Developers are to ensure that residual risks from constraints are not significant to harm or otherwise prevent the system from performing its functions, create unacceptable costs, or price the system's end products out of competitiveness. 2 Analyses of system requirements can necessitate consideration of existing or possible physical solutions to ensure feasibility.
e) Challenge questionable requirements	Acquirer and other stakeholder requirements that are of questionable utility or that have an unacceptable risk of satisfaction are identified and resolved.
f) Resolve identified conflict of requirements	Any conflicts between combinations of functional requirements, performance requirements, or constraints, as well as within respective sets of those requirements, are resolved. Invoked the Systems Analysis Process, Requirement 23.
g) Prepare a set of acceptable system technical requirement statements	Associated assumptions and technical requirement statements for the system are prepared and then validated. Invoked the Requirements Validation Process Requirement 25.
h) Ensure completeness and consistency of the set of system technical requirements	System technical requirements are validated. Resolution of variances is completed. Invoked the Requirements Validation Process, Requirement 28.
i) Record the set of system technical requirements	The validated set of system technical requirements and associated assumptions is captured in the project's information database and maintained and controlled throughout the life of the project. NOTE—Controlled maintenance of the system technical requirements in the information database allows for traceability, supports validation, and is essential for change management.

Table C.17—Requirement 17 (Solution Definition Process - Logical Solution Representations)

Representative tasks	Expected outcomes
a) Select and implement one or more of the four approaches below, or other approach designated by enterprise policies, guides, or standards: 1) Functional analysis 2) Object-oriented analysis 3) Structured analysis 4) Information modeling 5) Other techniques	An abstract definition of the solution is provided in the form of: 1) functional flow, timelines, behaviors, data and control flows, states and modes, functional failure modes and effects. 2) objects encapsulating a partition and mapping of System Technical Requirements and characterized by services (behaviors, functions, and operations) provided and by encapsulated attributes (values, characteristics, and data)

c) Assign system technical requirements (including performance requirements and constraints)	System technical requirements (including performance requirements of a functional requirement and constraints) assigned to appropriate subfunctions, groups of subfunctions, objects, data structures, etc. NOTE—There can be unassigned system technical requirements after the tasks of Requirement 17 are completed (see the note under Requirement 17, Task c).
d) Identify, define, and validate derived technical requirement statements	Derived technical requirement statements prepared that (1) reflect requirements associated with defined logical solution representations from tasks a) and b), (2) constitute expansion of previously defined derived technical requirements into more detailed lower level requirements, (3) represent system technical requirement statements (such as range) that are not appropriate for logical solution representations but through analysis can be made more specific (such as fuel capacity, engine efficiency, and vehicle resistance), and (4) individually and as a set, are well formulated in accordance with Requirement 25).
e) Ensure completeness and consistency of logical solution representations	Logical solution representations and assumptions are validated. Resolution of identified variances is completed. Invoked the Validation Process Requirement 29.
f) Record logical solution representations and derived technical requirements	The following are captured in the information database: (1) the data generated, selected arrangements and sequencing, assignments of system performance requirements, and constraints, (2) the validated sets of logical solution representations, (3) the derived technical requirements, along with source rationale and assumptions, and (4) any unassigned system technical requirements [see the note under Requirement, Task 17 c)].

Table C.18—Requirement 18 (Solution Definition Process - Physical Solution Representations)

Representative tasks	Expected outcomes
a) Analyze logical solution representation sets, assigned system and derived technical requirements.	The following are determined: 1) which logical solution set or assigned requirement provides a requirement for an enabling product associated with development, production, test, deployment/installation, training, support/maintenance, or retirement/disposal. 2) which logical solution set or assigned requirement can best be accomplished manually or by facilities, material, or data. 3) which logical solution set or assigned requirement can best be accomplished by hardware, software, or firmware products (new or existing). Invoked the Systems Analysis Process, Requirements 22 and 23, as necessary.
b) Assign representations, derived technical requirements and unassigned system technical requirements to appropriate physical entities:	The appropriate sets of functions, groups of functions, objects, behaviors, derived technical requirements, etc., are assigned to appropriate physical entities (e.g., sensor, engine, power source, storage device, structural frame, communication device, and computer) that will make up a physical solution. NOTE—This assignment to physical entities and generation of alternative solutions composed of these entities is tightly coupled and iterative.
c) Generate and evaluate alternative physical solution representations by performing the following tasks: NOTE—Appropriate models (digital, hardware or software, or both, partial or complete) or prototypes are normally created to help avert risk, identify critical product characteristics and enabling product requirements, identify control requirements for product integrity, perform sensitivity analyses to establish design margins, provide quantitative performance assessments, and select preferred physical solution representation.	
1) Identify and Define Physical Interfaces	Physical interfaces (human, form, fit, function, data flow, and interoperability) among specific physical entities that make up each end product physical solution alternative, among end products that make up the system, among end products and enabling products, and along with end products and other interfacing systems, are identified and defined. Physical interfaces (internal to the system and external) among specific solutions selected for each physical entity that make up the selected physical solution are designed and described.
2) Identify and Analyze Critical Parameters	For each identified critical performance parameter (TPM), the variability and the sensitivity of each alternative physical solution to that variability are identified and defined.
3) Identify and assess physical solution options:	
(a) Technology requirements	The technological needs necessary to make each alternative solution effective, the risks associated with introduction of new or advanced technologies to meet requirements, and alternative lower-risk technologies that could be substituted for unacceptable higher risk technologies are identified and assessed.
(b) Off-the-shelf availability	The availability of off-the-shelf end products (non-developmental hardware or reusable software) are identified and assessed.
(c) Competitive considerations	The effect of design considerations to maintain or make a physical solution representation alternative competitive with potential or existing competitor products is identified and assessed.
(d) Failure modes, effects, and criticality	Further design efforts are identified that will be needed to accommodate redundancy and to support graceful degradation when the results of failure modes, effects, and criticality of failure analyses have an unacceptable or high criticality rating.
(e) Performance assessment	The degree to which the performance requirements are satisfied by each alternative physical solution is identified and assessed.

(f) Life cycle considerations	The degree to which producibility, testability, ease of deployment, installability, operability, supportability, trainability, and disposability are considered in each alternative physical solution is identified and assessed. Enabling product needs, requirements and constraints for the associated processes are identified, assessed, and defined.
(g) Capacity to evolve	The capacity of each alternative physical solution to evolve, or be reengineered, incorporate new technologies, enhance performance, increase functionality, or other cost-effective or competitive improvements, once solution end products are in production or in the marketplace, are identified and assessed. Limitations that can preclude the capability of the system to evolve are identified and documented.
(h) Make vs. buy	The advantages and disadvantages of making the products of the solution within the enterprise or going to an established supplier are identified and assessed.
(i) Standardization considerations	The advantages and disadvantages of using standardized end products, protocols, interfaces, etc., for the physical solution are identified and assessed.
(j) Integration concerns	The following are identified and assessed: (1) potential hazards to other systems, operators, or the environment; (2) built-in test and fault-isolation test requirements; (3) ease of access, ready disassembly, use of common tools, part count effect, advantage of modularity, standardization, and less need for cognitive skills; and (4) dynamic or static conflicts, inconsistencies, and improper functionality of the integrated products of the solution.
4) Perform systems analyses	Which physical solution option is best for each alternative solution representation, based on each option individually or in sets, (Requirements 22, 23, and 24) is determined.
d) Identify and define derived technical requirements	Derived technical requirement statements identified and defined that are (1) the consequence of design choices associated with the above tasks, (2) used to form alternative physical solution representations, as appropriate, and (3) individually and as a set (including physical interface requirements) well formulated (Requirement 25).
e) Select preferred physical solution	The preferred physical solution representation is selected, based on the results of an evaluation of each physical solution representation (Requirements 22, 23, and 24).
f) Ensure selected physical solution representation consistency	The selected physical solution representation is determined to be consistent with assigned logical solution representations, derived technical requirements, and the identified subset of unassigned system technical requirements [see the note under Requirement, Task 17 c)].
g) Record the outcomes of a) through g)	The following are captured in the information database: selected physical solution representation, along with selection rationale, assumptions, and outcomes from tasks a) through g).

Table C.19—Requirement 19 (Solution Definition Process - Specified Requirements)

Representative tasks	Expected outcomes
a) Fully characterize design solution	For each specific physical entity of the selected physical solution: hardware drawings and schematics, software design documents, parts lists, interface descriptions, procedural manuals, data or other applicable design descriptions, based on the requirements assigned to the selected physical solution and engineering life cycle phase exit criteria, are completed, as applicable.
b) Ensure design solution consistency	The defined design solution is verified as being consistent with the selected physical solution representation as described by its encapsulated requirements for the assigned logical solution representations, associated system technical requirements, and derived technical requirements. Invoked the Verification Process Requirement 30.
c) Specify requirements	System, subsystem, and interface specifications that describe the specified requirements (functional and performance requirements, and physical characteristics) are documented. Test requirements to ensure that end products satisfy their specified requirements are determined and included in the related specification, as appropriate to the engineering life cycle phase.
d) Record design solution and related specified requirements	The design solution work products, including the specified requirements, are captured and recorded in the established information database, along with all tradeoff analyses, design rationale, assumptions, and key decisions to provide traceability of requirements up and down the system structure.
e) Establish projects for development of enabling products	A project is established to engineer the enabling products associated with the processes for development, production, test, deployment/installation, training, support/ maintenance, and retirement/disposal. NOTE—The requirements for enabling products come from (1) user or customer or assigned requirements and other stakeholder requirements for the system, and (2) derived technical requirements for end products and their subsystems generated by tasks of the Solution Definition Process. Thus, initiation of enabling product development is dependent on the completion of the design solution for the system (building block) being engineered or reengineered.

C.4 Product Realization task outcomes

Table C.20—Requirement 20 (Implementation Process)

Representative tasks	Expected outcomes
a) Acquire products (Goods or Services)	Hardware, software, firmware end products, or composites of end products built or coded to their specified requirements, drawings or descriptive documents; or other needed physical entities [for example, trained personnel, certified facilities, special techniques (manual procedures or processes), manuals)] are acquired. Hardware items were (1) purchased off-the-shelf from a supplier or vendor; (2) fabricated in-house; or (3) from in-house, off-the-shelf supply. Software items were (1) purchased from a supplier or vendor; (2) coded in-house; or (3) reused.
b) Validate acquired products	Acquired products are validated that each acquired end product or aggregation of end products is in conformity with its user, customer, or assigned requirements. Invoked the End Products Validation Process, Requirement 33. NOTE—This validation is accomplished by the supplier as per the agreement or by the acquirer, with or without supplier participation. This validation includes product certification or acceptance testing, as appropriate.
c) Assemble/integrate validated end products	End products or aggregations of end products already validated are physically integrated or assembled into the required test article or the end product that will be verified and delivered to an acquirer.
d) Verify integrated end products	End products are verified that each end product of the system under development complies with its specified requirements. Invoked the System Verification Process, Requirement 31.
e) Verify enabling products for each associated process	Enabling products for production, test, deployment/installation, training, support/maintenance, and retirement/disposal, as appropriate, are verified that they will be ready to perform the support functions required by the system's end products. Associated processes are proofed, as applicable. Invoked the System Verification Process, Requirement 32.
f) Validate the verified end product	End products are validated prior to delivery to their acquirer, if required in the agreement, using the End Products Validation Process, Requirement 33.

Table C.21—Requirement 21 (Transition to Use Process)

Representative tasks	Expected outcomes
a) Acquire and put in place enabling products	Appropriate enabling products for supporting the Transition to Use Process are acquired and put in place.
b) Prepare end products for shipping or storage	In accordance with the agreement, (1) packing materials and containers are prepared, and (2) end products are packaged and appropriately labeled for either storage or delivery.
c) Store or deliver end products	End products awaiting shipping are appropriately stored or, in accordance with the agreement, delivered to intended usage sites in a condition suitable for application, use, installation, or integration with other end products or composites of end products.

C.5 Technical evaluation task outcomes

Table C.22—Requirement 22 (Systems Analysis Process - Effectiveness Analysis)

Representative tasks	Expected outcomes
a) Plan effectiveness analyses	A plan is prepared to include the purpose, objectives, execution and data collection requirements, schedule of tasks, availability of required resources, expected outcomes, and the general approach for required effectiveness analyses.
b) Analyze system cost effectiveness	For each alternative physical solution representation, as well as for the design solution, the system cost effectiveness is determined with respect to the following attributes, as applicable: accuracy, availability, capacity, maintainability, reliability, responsiveness, operability, safety, security, survivability, spare requirements, transportability, vulnerability, etc.
c) Analyze total ownership cost	Costs to the enterprise and to the acquirer for alternative physical solution representations, for alternative tradeoff analysis options, or for proposed changes, and the known uncertainties (risks) in these costs are determined. NOTE—The following costs are typically included in a total ownership cost analysis: development, production, test, deployment/installation, training, operations, support/maintenance, and retirement/disposal.
d) Analyze environmental impacts	Applicable federal, state, municipal, and international environmental statutes and applicable hazardous material lists affecting the project and endurance of compliance by each physical solution are determined; the effect on and by each end product and enabling product on the infrastructure, land and ocean, atmosphere, water sources, and animal, plant and human life, as applicable, has been determined, from an enterprise-based life cycle perspective.
e) Analyze system effectiveness	For each operational profile, each alternative physical solution representation and the design solution are assessed by analytic confirmation to satisfy appropriate requirements.
f) Record outcomes of effectiveness analyses	Effectiveness analysis outcomes, as well as the details of the analyses performed, including rationale, assumptions, and lessons learned, are captured and recorded in the established information database.

Table C.23—Requirement 23 (Systems Analysis Process - Tradeoff Analysis)

Representative tasks	Expected outcomes
a) Plan tradeoff analysis	A plan is prepared to include: 1) the availability of required resources, level of importance, execution and data collection requirements, expected outcomes, objectives, schedule of tasks, and the type. NOTES—The types of tradeoff analyses typically performed include: 1 Formal—formally conducted, with results reviewed at technical reviews. Specific formal tradeoff analyses are normally identified in an agreement. 2 Informal—follows the same methodology of a formal tradeoff analysis but requires less documentation and is of less importance to the acquirer. 3 Judgmental—selection of a recommended option, based on judgment of the analyst or designer after a less rigorous analysis than that required by a formal tradeoff analysis and for which the consequences are not too important. One option is clearly superior to others or time is not available for a more formal approach. Most tradeoff analyses performed for engineering a system are of the judgmental type. 2) selection criteria that characterize what makes a specific option desirable or undesirable, such as (1) cost, schedule, performance, and risk; (2) life-cycle concerns; (3) -ility concerns (e.g., producibility, testability, maintainability, supportability, disposability); (4) size, weight, and power consumption for the type of tradeoff analysis selected; and (5) effectiveness analysis outcomes. 3) weighting factors for each criterion that will help distinguish its degree of importance for the defined tradeoff analysis. 4) applicable models (representative or simulation) that will support conduct of the tradeoff analysis, as well as determination that the model selected is valid for the tradeoff analysis to be performed. 5) list of viable optional solutions or courses of action to be evaluated.
b) Perform tradeoff analysis	Tradeoff analyses are completed according to the plan, with determination of: 1) quantitative basis for evaluating the tradeoff analysis options from appropriate effectiveness analysis tasks (Requirement 22). 2) quantitative assessment of the risk level associated with each option from appropriate risk analysis tasks (Requirement 24). 3) collection of data pertaining to each option evaluated and analysis of the data to determine the effect of each option on the system or project if implemented. Determination that the methodologies and data collection were sufficient to support a fair and complete evaluation. 4) identification and definition of the recommended option, based on the comparison of each option and its effects against the established success criteria. 5) presentation of the recommendations to the appropriate decision makers, as applicable.
c) Record outcomes of tradeoff analysis	Recommendations and the selection, as well as the details of the tradeoff analysis performed, including rationale, assumptions, and lessons learned, are captured and recorded in the established project information database.

Table C.24—Requirement 24 (Systems Analysis Process - Risk Analysis)

Representative tasks	Expected outcomes
a) Identify risks	Technical risks, and resulting project risks, are identified, based on exposure to the probability of an undesirable consequence and the effect of that consequence for each tradeoff analysis option or each physical solution representation option. Considerations include how expectations from a decision or design selection are affected by (1) commitments resulting from a choice, (2) validity of assumptions, (3) capabilities to implement and control, and (4) other organizational or technical constraints such as resources and time.
b) Characterize risks	Risk causes, possible effects or consequences, likelihood of occurrence, options for dealing with identified risks, how long options are available, and coupling among identified risks are determined.
c) Prioritize risks	Risks that would likely cause harm, would have the greatest effect, and would need immediate attention are prioritized.
d) Evaluate ways to avert risks	The cost, schedule, and performance effects on the project are determined from evaluation of options or courses of action that would (1) eliminate a specific risk possibility; (2) implement acts to reduce a risk's probability or effect; (3) transfer the risk (get someone else to assume the risk, e.g., a warranty); or (4) provide a contingency to address the consequences, if the risk occurs, including identity of appropriate and timely triggers for taking action (will they give sufficient time to act?) such as metrics or events to monitor.
e) Define and implement a plan or approach for averting each significant risk	The significant risks to the project are identified and adequate risk aversion approaches are defined. Triggers are defined that will provide a signal when it is appropriate to implement aversion action. Implemented planned actions or approach to avert risk.
f) Capture and communicate risk analysis outcomes	The effects of the risk analysis, as well as the details of the risk analysis performed, including assumptions, are captured and recorded in the established project information database. Risks effects have been reported or used, as appropriate.

Table C.25—Requirement 25 (Requirements Validation Process - Requirement Statements Validation)

Representative tasks	Expected outcomes
a) Analyze and ensure each technical requirement statement is stated with: 1) ability to preserve competitiveness—permits preservation of a competitive stance and is only as constraining on competitive stance as is justified by benefits delivered by requirement. 2) clarity—requirement statement is readily understandable without analysis of meaning of words or terms used. 3) correctness—requirement statement does not contain an error of fact. 4) feasibility—requirement can be satisfied within (1) natural physical constraints, (2) state of the art as it applies to the project, and (3) all other absolute constraints applying to the project. 5) focus—requirement is expressed in terms of 'what' and 'why,' or form, fit and function, not in terms of how to develop the products or the materials to be used — detailed requirements that are required to guide detailed design of a product are an exception to this. 6) implementability—requirement statement contains information necessary to enable requirement to be implemented. 7) modifiability—necessary changes to a requirement can be made completely and consistently. 8) removal of ambiguity—allows only one interpretation for meaning of the requirement, e.g., not defined by words or terms such as 'excessive,' 'sufficient,' and 'resistant' that cannot be measured. 9) singularity—requirement statement cannot be sensibly expressed as two or more requirements having different agents, actions, objects, or instruments. 10) testability—existence of finite and objective process with which to verify that the requirement has been satisfied. 11) verifiability—can be verified at the level of system structure at which it is stated.	
b) Analyze and ensure technical requirement statements in pairs and as a set are stated with: 1) absence of redundancy—each requirement is specified only once. 2) connectivity—all terms within a requirement are adequately linked to other requirements and to word and term definitions, so that individual requirements relate properly to other requirements as a set. 3) removal of conflicts—requirement is not in conflict with other requirements or within itself.	

Table C.26—Requirement 26 (Requirements Validation Process - Acquirer Requirements)

Representative tasks	Expected outcomes
a) Select methods and define procedures	The methods and procedures for validating the set of defined acquirer requirements are selected and defined, consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.
b) Establish downward traceability	The downward traceability of stated, documented, or otherwise determined, acquirer needs and expectations to the set of defined acquirer requirements is determined.
c) Establish upward traceability	The upward traceability of the individual acquirer requirements, from the set of defined acquirer requirements, to stated, documented, or otherwise captured, acquirer needs and expectations is determined.
d) Identify and resolve variances	Identified voids, variances, and conflicts have been resolved. When the set of defined acquirer requirements is not upward-traceable to acquirer needs and expectations, whether non-sourced (orphaned) requirements or constraints were introduced and whether they are desired by the acquirer, have been determined, and appropriate action has been taken. When acquirer needs and expectations are not reflected in the set of defined acquirer requirements, the omitted needs and expectation are added to the set of defined acquirer requirements, as appropriate.
e) Record validation results	Validation procedures, outcomes, assumptions, corrective actions, lessons learned, etc., are captured and recorded in the established information database.

Table C.27—Requirement 27 (Requirements Validation Process - Other Stakeholder Requirements)

Representative tasks	Expected outcomes
a) Select methods and define procedures	The methods and procedures for validating the set of defined other stakeholder requirements are selected and defined and are consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.
b) Establish downward traceability	The downward traceability of stated, documented, or otherwise determined, other stakeholder needs and expectations to the set of defined other stakeholder requirements is established.
c) Establish upward traceability	The upward traceability of the individual other stakeholder requirements, from the set of defined other stakeholder requirements, to stated, documented, or otherwise captured, other stakeholder needs and expectations is established.
d) Identify and resolve variances	Identified voids, variances, and conflicts are resolved. When the set of defined other stakeholder requirements was not upward-traceable to other stakeholder needs and expectations, whether non-sourced (orphaned) requirements or constraints were introduced, has been determined, and appropriate actions were taken to eliminate non-sourced requirements. When other stakeholder needs and expectations were not reflected in the set of defined other stakeholder requirements, omitted needs and expectation were added to the set of defined other stakeholder requirements, as appropriate.
e) Record validation results	Validation procedures, outcomes, assumptions, corrective actions, lessons learned, etc., are captured and recorded in the established information database.

Table C.28—Requirement 28 (Requirements Validation Process - System Technical Requirements)

Representative tasks	Expected outcomes
a) Select methods and define procedures	The methods and procedures for validating the set of defined system technical requirements are selected and defined and are consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.
b) Establish downward traceability	The downward traceability of the validated sets of stakeholder (acquirer and other stakeholder) requirements to the set of defined system technical requirements is determined.
c) Establish upward traceability	The upward traceability of the individual system technical requirements, from the set of defined system technical requirements, to the validated sets of stakeholder requirements is determined.
d) Analyze assumptions	Assumptions regarding consistency of the system technical requirements with the system being engineered are determined.
e) Analyze other system technical requirements	Other system technical requirements derived as essential to design and subsequent life cycle phases are consistent with the system being engineered and other system technical requirements are determined.
f) Identify and resolve variances	Identified voids, variances, and conflicts are resolved. When the set of defined system technical requirements was not upward-traceable to validated sets of stakeholder requirements, whether non-sourced (orphaned) requirements or constraints were introduced was determined, and appropriate actions to eliminate non-sourced requirements or revise the appropriate set of stakeholder requirements were taken. When validated stakeholder requirements were not reflected in the set of defined system technical requirements, omitted requirements were added to the set of defined system technical requirements or determine the need for the requirement, as appropriate.
g) Perform revalidation	When a change is needed to one of the validated sets of stakeholder requirements, the appropriate tasks of acquirer or other stakeholder requirements definition from the Requirements Definition Process were reaccomplished and the set was revalidated. When the set of system technical requirements must be changed, the appropriate tasks of system technical requirements definition from the Requirements Definition Process were reaccomplished and the set was revalidated.
h) Record validation results	Validation procedures, outcomes, assumptions, corrective actions, lessons learned, etc., are captured and recorded in the established information database.

Table C.29—Requirement 29 (Requirements Validation Process - Logical Solution Representations)

Representative tasks	Expected outcomes
a) Select methods and define procedures	The methods and procedures for validating the defined sets of logical solution representations and derived technical requirements are selected and defined and are consistent with the level of system structure, enterprise-based life cycle phase, and Validation Plan, as appropriate.
b) Establish downward traceability	The downward traceability of the validated set of system technical requirements to each set of logical solution representations and the derived technical requirements is determined.
c) Establish upward traceability	The upward traceability of individual logical solution representations from a set of logical solution representations and the derived technical requirements to the validated set of system technical requirements is determined.
d) Analyze assumptions	Assumptions made while defining the sets of logical solution representations to ensure that they are consistent with the system technical requirements and the system being engineered are assessed and considered valid.
e) Identify and resolve variances	Identified voids, variances, and conflicts are resolved. When validated system technical requirements were not reflected in a set of logical solution representations, omitted requirements were added to the set of logical solution representations. The need for added requirements was confirmed, and it was determined whether these requirements were to be assigned directly to physical solutions. When a set of logical solution representations was not upward traceable to the validated set of system technical requirements, it was determined whether non-sourced (orphaned) requirements and constraints had been introduced. Appropriate actions were taken either to eliminate non-sourced requirements, to establish derived requirements, or to revise the set of system technical requirements.
f) Perform revalidation	When a change was needed to the validated set of system technical requirements, the appropriate tasks from the Requirements Definition Process were reaccomplished and the set was revalidated. When one or more sets of logical solution representations had to be changed, the appropriate tasks for definition of logical solution representations from the Solution Definition Process were reaccomplished and the set was revalidated.
g) Record validation results	Validation procedures, outcomes, assumptions, corrective actions, lessons learned, etc., are captured and recorded in the established information database.

Table C.30—Requirement 30 (System Verification Process - Design Solution Verification)

Representative tasks	Expected outcomes
a) Plan the design solution verification in accordance with the Verification Plan, the agreement, and the applicable enterprise-based life cycle phase, and level in the system structure	1) The appropriate method needed to verify the system's fully characterized design solution is identified and defined. NOTE— Design solution verification methods include**:** inspection (for example, inspection of drawings), analysis (for example, using simulation or virtual reality prototype), demonstration (for example, using mockups or physical models), or test (for example, by testing physical prototypes, breadboards, or brassboards). 2) Verification procedures are defined, based on (1) procedures for each method selected, (2) purpose and objective of each procedure, (3) pre-test and post-test actions, and (4) criteria for determining the success or failure of the procedure. 3) The verification environment (for example, facilities, equipment, tools, simulations, measuring devices, personnel, and climatic conditions) in which the verification methods and procedures will be implemented is established and checked-out for adequacy, completeness, readiness, and integration.
b) Perform the planned design solution verification using selected methods and procedures within the established verification environment	Verification outcomes to show completion of verification objectives and to determine untraceable requirements and constraints, voids, conflicts, variations and anomalies are collected and evaluated. Specifically, it was shown that: 1) the system design solution descriptions and interfaces (internal or external) are upward-traceable to requirements of the selected physical solution representation; 2) source requirements are downward-traceable to the system design solution descriptions; 3) the design solution satisfied the functional and performance requirements of the identified subset of unassigned system technical (see note under Requirement, Task 17 c) and the set of derived technical requirements; 4) intended functions are correctly implemented; 5) constraints, including interfaces, are satisfied. When defined variances were not downward-traceable from source documents, appropriate tasks of the Requirements Definition and Solution Definition Processes were repeated to correct the omissions. When defined variances showed inconsistencies with source requirements (not upward-traceable), the followed were determined: why new requirements were introduced, and if they were to be assigned as derived technical requirements, were to be removed from the design solution definition, or had to be reflected in the set of logical solution representations or set of system technical requirements. The necessary tasks of the Requirements Definition and Solution Definition Processes were reaccomplished as required for corrections and reverification.
c) Perform reverification	When test outcome variations and anomalies were traced to poor verification conduct or to inadequate verification environment, verifications are repeated to obtain valid outcomes.
d) Record verification results	The verification procedures, together with the outcomes achieved, variations, corrective actions taken, rationale justifying the design solution, tradeoff analyses and effectiveness analyses completed with resulting key decisions, verified design solution definition, lessons learned, etc., are recorded in the project information database according to the verification plan and test procedure requirements. NOTE—The verified design solution and its related specified requirements are placed under configuration management control.

Table C.31—Requirement 31 (System Verification Process - End Product Verification)

Representative tasks	Expected outcomes
a) Plan the end product verification in accordance with the Verification Plan, the agreement, and the applicable enterprise-based life cycle phase, and level in the system structure.	1) The appropriate methods needed to verify the system's end products against their specified requirements are selected and defined. NOTE— Design solution verification methods include**:** inspection (for example, inspection of drawings), analysis (for example, using simulation or virtual reality prototype), demonstration (for example, using mockups or physical models), or test (for example, by testing physical prototypes, breadboards, or brassboards). 2) Verification procedures are established and based on (1) procedures for each method selected, (2) purpose and objective of each procedure, (3) pre-test and post-test actions, and (4) criteria for determining the success or failure of the procedure. 3) The verification environment (for example, facilities, equipment, tools, simulations, measuring devices, trained personnel, special techniques, and climatic conditions) in which the verification methods and procedures will be implemented is established and checked out for adequacy, completeness, readiness, and integration. 4) Test articles are on hand, assembled, and integrated with the verification environment according to verification plans and schedules, and appropriate sets of specified requirements are available.
b) Perform the planned end product verifications using selected methods and procedures within the established verification environment.	Verification outcomes are collected and evaluated to show completion of verification objectives and used to determine 1) variations and anomalies, and out-of-compliance conditions; 2) data quality, integrity, correctness, consistency, and validity; 3) whether fabricated, integrated, or purchased end products (including end products, composites of end products, or software or firmware builds) comply with their respective specified requirements; 4) that end product test articles were appropriately integrated with the test environment and each requirement was properly tested for; and 5) that system end products function together and with interfacing products throughout their performance envelope. For variations and anomalies not caused by poor test conduct or conditions, appropriate tasks of the processes in this Standard, including replanning, changing requirements, redefining requirements and the design solution, and verification, are accomplished to resolve discrepancies.
c) Perform reverification	When test outcome variations and anomalies were traced to poor verification conduct or conduct or inadequate verification environment, end product verification is reaccomplished.
d) Record verification results	The verification methods and procedures, together with the outcomes achieved, variations and anomalies, corrective actions taken, rationale justifying corrections, tradeoff analyses, and effectiveness analyses completed with resulting key decisions, lessons learned, etc., are recorded in the project information database according to the verification plan and test procedure requirements. Recorded test result data includes the following: 1) The version of the set of specified requirements (specification) used. 2) The version of the end product tested. 3) The version or reference standard for tools and equipment used, together with applicable calibration data. 4) The results of each test including pass or fail declaration. 5) The discrepancy between expected and actual results. 6) A statement of success or failure of the testing process, including its relation to the verification process. Delivery or disposition of verified compliance articles and compliance data is completed in accordance with the acquirer-supplier agreement, verification plan instructions, or project directives or procedures.

Table C.32—Requirement 32 (System Verification Process - Enabling Product Readiness)

Representative tasks	Expected outcomes
a) Plan enabling product readiness determination in accordance with the agreement, the applicable enterprise-based life cycle phase, and level in the system structure	1) The appropriate methods needed to determine enabling product readiness and maturity of development, based on the applicable enterprise-based life cycle phase and level in the system structure, the purpose and objective of each method selected, the appropriate plan, and the acquirer-supplier agreement, are selected and defined. 2) Procedures based on (1) each method selected, (2) purpose and objective of each method, (3) pre-test and post-test actions, and (4) criteria for determining the success or failure of the method are established. 3) The environment (for example, facilities, equipment, tools, simulations, measuring devices, trained personnel, special techniques, and climatic conditions) in which the methods and procedures will be implemented is established and checked out for adequacy, completeness, readiness, and integration. 4) Required information regarding the status and maturity of enabling product development or requirements definition is on hand. Non-developmental enabling products are on hand and integrated appropriately.
b) Perform planned enabling product readiness determination, using selected methods and procedures	Outcomes are collected and evaluated, and any enabling product readiness anomalies, variations, or out-of-compliance conditions (such as lack of requirements for manuals or training equipment or disposal of hazardous materials) are discovered. The following have been determined 1) whether development for required enabling products is progressing satisfactorily or will be ready to perform its life cycle function when needed or if there are out-of-compliance conditions 2) that plans and selected methods, procedures, and tools for each associated process can accomplish their intended purpose 3) whether the development is on schedule and that the schedule meets critical end product needs 4) the interfaces between planned enabling products and their intended end products have no potential conflicts in implementation concepts, intended functions, or interdependencies 5) that enabling products meet the requirements of the end products or composites of end products they are intended to support. For variations and anomalies not caused by poor readiness assessments, appropriate tasks of the processes in this Standard, including replanning, changing requirements, redefining requirements and the design solution, and readiness determination, are accomplished to resolve discrepancies.
c) Reaccomplish readiness determination	For discrepancies caused by poor readiness assessment, the appropriate tasks of enabling product readiness determination are reaccomplished.
d) Record readiness determination results	Enabling product readiness determination outcomes are recorded in the information database.

Table C.33—Requirement 33 (End Products Validation Process)

Representative tasks	Expected outcomes
a) Determine validation exit criteria	The type of validation required and the requirements to be used are determined. The types include: (1) validation against acquirer requirements in the anticipated usage environment, with test conditions that span the expected range of actual operating conditions, to the extent practical, and in conjunction with stakeholders, as appropriate; (2) certification tests against established certification requirements; (3) acceptance tests using operational processes and personnel in operational environments; or (4) as specified in the agreement. NOTES 1 Validation tests are conducted during the Test and Evaluation Phase of the engineering life cycle, after end products have been verified against specified requirements, from the lowest level of the system structure upward to the end products that will be delivered to the marketplace to satisfy validated acquirer requirements. 2 Validations of Types 1 through 3 are satisfied with the same test, when appropriate. 3 Validation can be for a single end product or an aggregation of end products for the same building block.
b) Acquire appropriate test article	The test article, or test articles, used for the validation is determined to be appropriate to the enterprise-based life cycle phase and the level of system structure. NOTE—End Products Validation consists of one or more tests using a version of the product (or products) as nearly like the final version as is practical and necessary, taking into account the enterprise-based life cycle phase and the nature of the product. If the nature of either the product, its operating conditions, or the enterprise-based life cycle phase of development precludes use of actual products or prototypes, then breadboards, brassboards, hardware-in-the-loop simulations, virtual-reality simulations, or other models and simulations are applicable for End Products Validation.
c) Conduct validation	1) Validation is completed in accordance with the Validation Plan, as required in the agreement. 2) Validation outcomes are compiled, analyzed, and compared to the validation exit criteria; variations and anomalies have been identified; and corrective actions are defined. 3) When outcome variances from exit criteria were not caused by improper test conditions, by improper performance of validation procedures, or by improper data collection: Replanning, redefinition of the design solution, and the Implementation Process, as appropriate, are reaccomplished. NOTE—Care is to be taken to ensure that the requirements derived to remove variances do not conflict with acquirer or other stakeholder requirements, or other validated technical requirements without coordinating such changes with the appropriate stakeholders.
d) Perform revalidation	If variances were caused by poor test conduct, retesting, using improved or correct test equipment and procedures, is performed.
e) Record validation results	Validation procedures, compliance data, outcomes, assumptions, corrective actions, lessons learned, etc., are recorded in the established project information database.

Annex D—Planning documents (informative)

This Annex provides an informative list of typical documents and their contents taken from various commercial and non-commercial domains. Selection and use of these documents depends on agreement requirements and the nature and scope of the project.

D.1 Source documents

1. During early phases of the enterprise-based life cycle, system concepts are often vague and unstructured. Typical concept source documents include:

   a) *Concept Specification*. This includes a features list for a new or improved system or product. It identifies the scope of the features and their priority to provide an edge in the market.

   b) *Maintenance Concept*. This focuses on life cycle logistics goals, objectives, constraints, and general support capabilities related to a desired system or product.

   c) *Operations Concept (or Concept of Operations)*. This focuses on the goals, objectives, and general desired capabilities of a potential system or product (new or improved), without indicating how the system or product can be implemented.

   d) *Disposal Concept*. This focuses on the planned disposition of the end products, and by-products produced throughout the life cycle of the system.

   e) *Request for Proposal (RFP*). This can include one or more of the above initiating documents. Its purpose is to solicit bids for consideration from several sources to develop a system or product.

2. For creation activities (system definition, subsystem design, detailed design and integration, and test and evaluation) of the engineering life cycle, source documents are much more definitive and include one or more of the following:

   a) *Contract*. This type of negotiated document is the basis for most project efforts involving two enterprises. It often includes the Operations Concept, Maintenance Concept, Statement of Work, performance specifications, drawings, and interface control documents.

   b) *Statement of Work (SOW)*. This provides requirements for the technical work to be accomplished by an assigned team or project. It is provided as part of a contract or internal tasking document.

   c) *Tasking Document*. This is a type of an agreement between two parties, typically inside an enterprise.

D.2 Technical documents

Technical documents are dependent on the applicable enterprise-based life cycle phase and describe the technical efforts in a particular area to be accomplished during engineering life cycle activities. Technical documents are usually prepared by the project during an earlier enterprise-based life cycle activity. They also **can** be included in source documents when prepared by the acquirer, either internal or external. Technical documents include (alphabetically):

1) *Configuration Management (CM) Plan.* This document defines the process used to identify and document the functional and physical characteristics of the system during its life cycle. The CM process provides a means of controlling changes to those characteristics and provides information on the status of changes. (See ANSI/EIA-649, *National Consensus Standard for Configuration Management.*)

2) *Contractor Integrated Technical Information Services (CITIS) Plan.* This document describes the methods that allow access and delivery of required digital information to an external acquirer.

3) *Data Management Plan.* This document reflects the data requirements of an agreement; establishes data management criteria and responsibilities; and describes the enterprise structure, administration, and control procedures used to ensure effective data management (internally as well as with external acquirers or suppliers).

4) *Electromagnetic Compatibility/Interference (EMC/EMI) Control Plan.* This document presents the methods that allow the project to meet the EMC/EMI requirements related to the system, including, as appropriate, susceptibility to electromagnetic pulse from nuclear weapons. It communicates the work effort, the emphasis, and the design guides to be used in avoiding serious electromagnetic compatibility problems. It provides guidance to assigned teams on design, specifications, and installation parameters so as to ensure a system that is compatible with upper-layer and lateral end products and enabling products, and with external systems.

5) *Engineering Plan*. The engineering plan provides, to project personnel and the acquirer, the planned technical efforts to accomplish the processes for engineering a system for the applicable enterprise-based life cycle phases of the project. The engineering plan provides: (1) an understanding of the problem to be solved, (2) what is planned to be accomplished, (3) how it will be done, (4) who will do it, (5) where and when things will be done, and (6) resources required, including when, how much, and characteristics. The focus of this plan is on risk reduction. This plan need not be a stand-alone document but can be part of the project plan. In military projects, this plan is often called the *Systems Engineering Management Plan* (SEMP).

6) *Human Factors/Engineering Plan.* This document focuses on human factors engineering so that the best human performance is obtained in the operation of the highly complex equipment developed by a project. This plan is built on the assumption that the capacities of humans lie within certain limits and that by adapting the design of end products to these capabilities, a more effective system results. A successful design of an end product for humans requires consideration of basic human characteristics: decision-making capability; muscular strength and coordination; body dimensions; perception and judgment; skills; optimum work load; and requirements for safety, comfort, and freedom from environmental stress.

7) *Interface Control Plan.* This document identifies and defines the physical, electronic, and content characteristics of all system internal and external interfaces and communications links. It ensures that the various elements of the system are functionally, physically, and electronically capable of interacting with each other, and with all external links with which they must connect or communicate, to perform required functions. This includes interfaces with people as well as hardware and software.

8) *Supportability Plan.* This document is meant: to influence the end product design solution definition activities to consider supportability requirements; to identify the support problems

and items that drive the cost of support early enough to change the design to fix or eliminate the support problems; to develop a complete set of projections of all resources required to support the end products over their life time; and to develop and use a single database for all analysis.

9) *Maintenance Plan.* This document emphasizes: understanding system readiness and performance requirements, physical environments, and resource availability to support the mission and purpose of the end products; managing the contributions to end product maintainability that are made by enabling products; developing robust end products that are insensitive to the environment experienced throughout the end product's life cycle and that are easily repaired under adverse conditions; and determining spares requirements.

10) *Producibility Plan.* This document has as its objective the achievement of a producible design solution definition at the lowest possible cost while maintaining the functional integrity and quality standards of system products. It includes planning for the analysis and coordination of both internal-supplier and external-supplier engineering, manufacturing, and procurement, and provides an orderly transition from development to production. Producibility emphasizes elimination of undesirable production features involving number of parts, materials, raw material forms, fabrication processes, tooling, and facilities.

11) *Reliability Plan.* This document has as its purpose the prevention, detection, and correction of design anomalies, weak parts, and workmanship defects.

12) *Software Development Plan*. This document describes a developer's plan for conducting a software development effort, whether for a new development, modification, reuse, reengineering, maintenance, or for all other activities resulting in software products. It includes the software development process to be used, the activities to be performed in each software build, and methods to be used.

13) *Specifications.* As a function of engineering life cycle activities, two kinds of specifications can be available—performance specifications and detail specifications. Performance specifications are outputs of the Solution Definition Process during the Pre-System Definition phase of the engineering life cycle and at least through the Subsystem Design phase. Performance specifications generally are stated in form, fit, and function terms. They can designate the means for verifying compliance. Detail specifications are typically an output of the detailed design activity, especially during development of lower-layer building block product designs. Detail specifications generally state requirements, characteristics, and materials related to a specific solution or approach, thus reducing developer flexibility. Both kinds of specifications can be included in a government contract or can be provided by the user, prime contractor, or another project.

14) *System Safety Plan.* This document has the objective of identifying, evaluating, eliminating, or controlling hazards throughout a product's life cycle. This plan is used to increase safety awareness within assigned teams and to design safety into end products.

15) *System Security Plan.* This document has the objective of identifying, evaluating, eliminating, or controlling security concerns. This plan is used to increase security awareness and bring about the design of security features that will a) reduce an organization's liability, b) address privacy issues, and c) correctly assist in preserving system operations and maintain system integrity when accidental or malicious fault events occur.

16) *Testability Plan.* This document is the basic tool for establishing and executing an effective testability program. This plan emphasizes: integration of testability requirements with

other design requirements and dissemination to assigned teams and external suppliers; establishing control for ensuring that each supplier's testability practices are consistent with end product requirements; identifying testability design guides and testability analysis models and procedures to be used by teams; planning for review, verification, and use of testability data submissions; and establishing the testability tasks that are to be done, how each task is to be done, when they are to be done, and how the results of the tasks are to be used.

17) *Training Plan.* This document establishes the personnel and training requirements; describes the supplier-provided training courses by type to establish skill levels to effectively perform operations and support activities; and identifies resources and supporting actions required for establishment and support of the training courses.

18) *Other technical plans*. The above list of technical plans is not meant to be exhaustive. This Standard calls for other plans such as *Verification Plans*, *Validation Plans*, and *Test Plans* for which much of the information in the *Testability Plan* would be included for any one or a group of specific tests; and a *Technical Performance Measurement Plan* (TPM). Other technical plans that can be applicable to a project include: *Computer Resource Development Plan, , Manufacturing Plan, Mass Properties Management Plan,* and *Test and Evaluation Master Plan* (TEMP),

D.3 Enterprise or project documents

Enterprise or project documents provide directive and constraining inputs to the Planning Process. These documents include:

1) *Enterprise Policies.* Policy documents provide a framework for decision making in the conduct of a project and the engineering of systems. Policies establish the criteria by which decisions are made in planning particular areas of an engineering effort as well in implementing an engineering effort. For instance, a policy could state that this Standard must be used for planning all enterprise project activities; or, that engineering efforts are to be accomplished using teams within the project-organizing structure; or, that projects are to use a particular automated tool to accomplish a certain task or set of tasks within the processes for engineering a system; or, the frequency of reporting progress or making progress checks.

2) *Enterprise or Project Procedures.* Procedure documents contain the recommended processes, approach, or steps to be taken in completing an agreement for engineering a system. Examples of procedures are: how reports are approved; or, how technical reviews are planned, conducted, and closed; or, the activities involved with planning, conducting, and reporting qualifying tests or validations.

3) *Project Plan.* This document provides the considered management approach to meet the requirements of an agreement. It lays out resource availability as a function of time and other key development schedule requirements. It also provides the budget over the projected time period of the applicable enterprise-based life cycle phases and policies for managing suppliers. This plan establishes the necessary boundaries for the engineering plan and other technical plans. In military projects, this plan often takes the form of an Integrated Master Plan (IMP) and Integrated Master Schedule (IMS).

4) *Resource Management Plan.* This document can contain: staffing availability, manpower loading limitations, delivery schedule dates, facility availability dates, capacity restrictions, and use of particular materials or reusable hardware or software units. These constraints

provide process and design limits, based on enterprise and project resource availability or policies.

5) *Risk Management Plan*. This document describes the project aspects of risk identification (sources and causes), risk characterization (effects, probabilities, choices, time frame, and coupling), risk prioritization (greatest harm, greatest effect, and time urgency), and risk aversion (mitigation, avoidance, transfer, and acceptance). It identifies the risk management functions to be performed by assigned teams and by supporting analysts and specialists. The acceptable levels of risk for a particular enterprise-based life cycle phase, or group of phases, are included.

6) *Strategic Plan.* This document provides insight into the projects and the markets the enterprise plans to pursue over a given time frame. The Strategic Plan establishes the desired enterprise direction, key objectives, strategies for attaining the objectives, and metrics by which progress toward meeting objectives is measured. It presents how the enterprise plans to compete to obtain a competitive advantage to outperform competitors. Plans for an engineering effort are to be consistent with and support the strategic plan.

7) *Total Cost of Ownership Plan.* This document describes the time-phased technical efforts required to control the total ownership cost, and, hence, the affordability, of a system under development. The ultimate cost of a system and its products is locked-in very early in the enterprise-based life cycle and with each application of development life-cycle processes. This document, therefore, discusses the enterprise's or project's plan for equating cost with performance and schedule requirements in evolving the system design; for balancing the future costs of production, operation, support, training, and disposal; and for taking active measures for meeting affordability objectives. Specifically, the cost of personnel and consideration of system complexity, open system architectures, reuse, and other such cost-saving approaches are included in the plan.

Annex E—System technical reviews (informative)

The system technical reviews of Table E.1 are related to engineering life cycle phases and are relevant to the system element of applicable building block developments. They are not directly related to enterprise-based life cycle phases (see Annex B); however, technical review exit criteria include satisfying the exit criteria of the applicable enterprise-based life cycle phase.

System technical reviews for a building block development can be formal (i.e., required by the external customer agreement). Incremental technical reviews for the subsystems, associated processes, and end products are generally informal, not requiring external customer participation on the reviewing body, and are normally conducted prior to the system technical review. System technical reviews for lower-layer building block developments are generally informal unless required to be formal in an agreement.

Table E.1—System technical reviews

PHASE	ENGINEERING LIFE CYCLE PHASE REVIEWS
Pre-System Definition	An *alternative system review*, if applicable, considers all concepts looked at and selects a preferred concept for further development that has the potential for satisfying identified stakeholder requirements. Assesses progress toward converging on a viable, traceable set of system technical requirements that are balanced with cost, schedule, and risk.
System Definition	A *system requirements review* validates that the set of stakeholder requirements is complete, consistent with acquirer's intent, and understood by the developer. A *system definition review* demonstrates convergence on and achievability of technical requirements and readiness to initiate the Subsystem Design Phase.
Subsystem Design	A *subsystem requirements review*, held for each subsystem-layer building block development, validates that the set of assigned and other local stakeholder requirements is complete, consistent with stakeholder's intent, and understood by the developer. A *system preliminary design review* for each subsystem building block development confirms that a) subsystem building block specifications have been defined appropriately; b) subsystem building block end product designs satisfy requirements assigned from the parent building block; c) enabling products for the associated processes have been defined adequately to initiate enabling product developments; d) the approaches planned for next lower-layer building blocks are appropriately planned; e) lower-layer building block project risks are identified, and mitigation plans are feasible and judged to be effective.

Table E.1—System technical reviews (continued)

Detailed Design	A *system detailed design review* for each lower-layer building block development demonstrates that a) specifications and/or drawings or software development files have been appropriately defined; b) the building block end product designs satisfy requirements assigned from the parent building block; c) enabling products for the associated processes have been defined adequately to initiate enabling product developments; d) the building block project is either: 1) ready for continued development; 2) appropriately defined for purchase of products from an external supplier; 3) ready for fabrication of building block elements; or 4) adequately defined so that off-the-shelf products or reuse products can be used to fulfill product requirements and are available within the enterprise.
End Product Physical Integration, Test and Evaluation	*Readiness reviews* for each building block from the bottom up demonstrate that delivered end products from lower-layer building blocks have been validated, or that validation tests are adequately planned, and that each set of integrated products forming a composite end product is ready for end product verification and end product validation, if required. *Audits* are intended to a) demonstrate that end product verification is complaint with their specified requirements and confirms that product verification outcomes compare favorably against configuration documentation: 1) drawings; 2) test procedures; 3) authorized changes; 4) software development files; and 5) "as-built" or "as-coded" documentation; b) confirm that the "as-built" or "as-coded" configuration has been favorably examined against its configuration documentation: 1) drawings; 2) bill of materials; 3) specifications; 4) code lists; 5) manuals; 6) compliance test; or 7) compliance data. Additionally, *audits* confirm that a) products have been built to drawings and satisfy specifications; b) the information database represents the work products of the building block development; c) required changes to previously completed specifications have been implemented; d) enabling products for down-stream associated processes are available, can be executed, and meet stakeholder requirements. *Process reviews* demonstrate that development of enabling products for associated processes is on schedule, and that designs satisfy related end product needs. Production readiness reviews and test readiness reviews are examples of process reviews.

Annex F—Unprecedented and precedented development (informative)

This informative annex provides guidance on the application of the System Design processes to a building block when the end products are either precedented or unprecedented.

The number of applications of the System Design Processes necessary to fully define the end products of a building block is a function of whether or not the end products that are to be used to meet technical requirements are already known. Generally, if a building block consists of one or more unprecedented items, at least two applications of the System Design processes will be required. The application of the requirements of this standard to a building block that consists of an unprecedented end product is shown in Figure F.1.

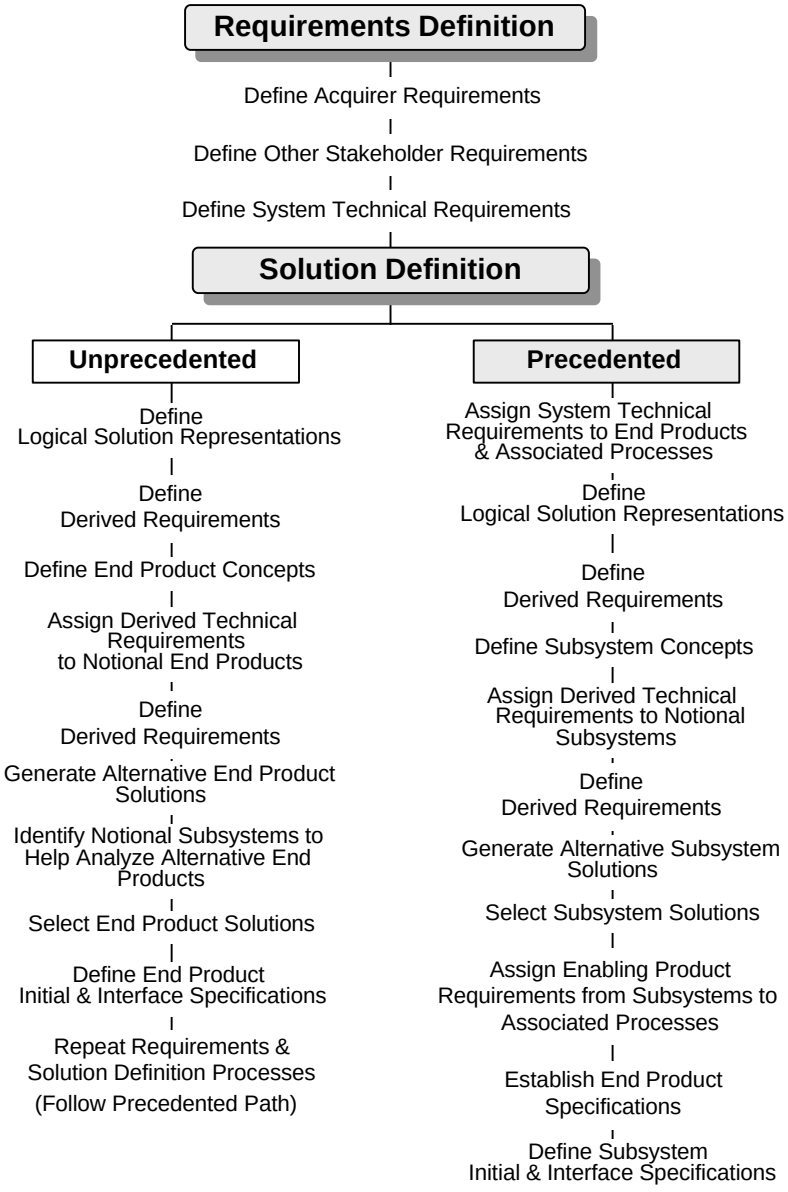

Figure F.1—Approaches for unprecedented and precedented end products

Annex G—Requirement relationships (informative)

The generation and use of various requirements and representations are introduced in Subclause 4.3. These are further described below. Figure G.1 shows the relationship of these requirements.

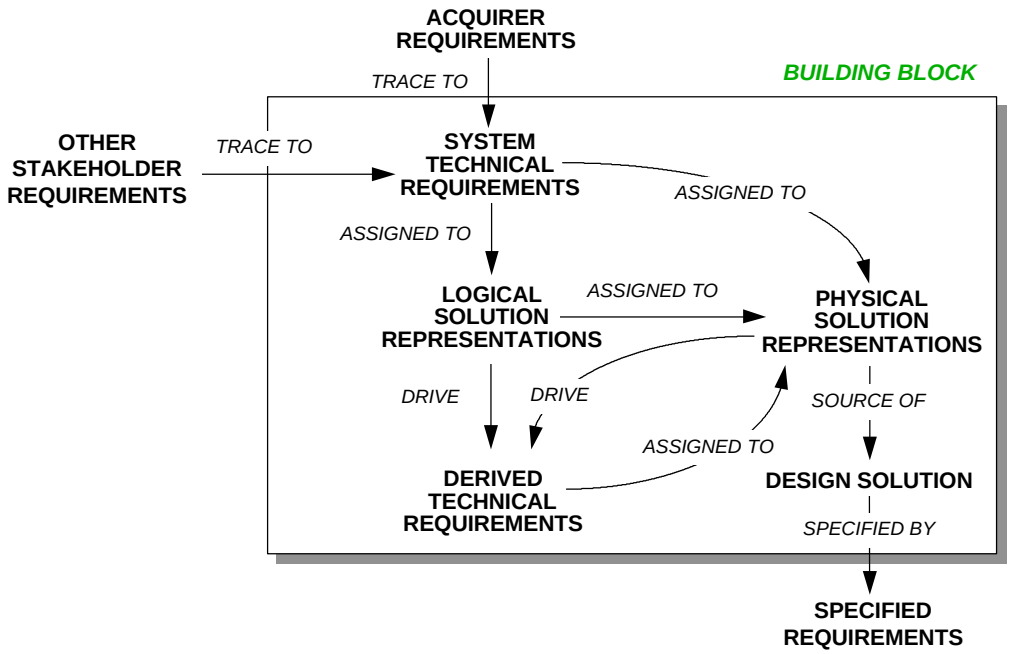

Figure G.1—Requirement relationships

Acquirer requirements come from a customer or user (including operators, where applicable) for a major system such as an aircraft, automobile, check processor, mail sorter, or telecommunication switch. Acquirer requirements also come from a developer needing subsystems to make up an end product of a system (see Subclause 6.2). The latter are identified as *assigned requirements* and would have been defined by a prior application of the System Design processes of Subclause 4.3.

Other stakeholder requirements, when added to the acquirer requirements, make up a set of stakeholder requirements that are transformed into *system technical requirements*. Stakeholder and system technical requirements are identified, collected, or defined by completing the Requirements Definition Process (Subclause 4.3).

The *logical and physical solution representations, derived technical requirements, design solution* and *specified requirements* are defined by completing the Solution Definition Process (Subclause 4.3).

Stakeholder requirements (acquirer and other stakeholder requirements), as well as system technical requirements and the derived technical requirements, differ from specified requirements.

1) In effect, stakeholder requirements constitute the input that establishes the problem to be solved. Such requirements can be considered as the initial specification for a development effort or as a set of specified requirements for procuring an off-the-shelf item. End products developed or purchased, and that are to be or that have been delivered to an acquirer, are validated against these specifications (see Requirement 33).

2) The derived technical requirements, logical solution representations, and system technical requirements reflect intermediate evolution states that are technical in nature, are validated, and are measurable. The design solution is verified against these requirements as reflected by the selected physical solution representation (see Requirement 30).

3) Specified requirements constitute the controlled definition of the finished solution. These requirements have two roles (see Figure G.2). The first role is to represent the build-to, buy-to, or assemble and integrate-to specifications, drawings, parts lists, etc., that describe the design solution of the end product. The associated specifications are those against which the end product will be verified (see Requirement 31). The second role is to represent the assigned requirements to be used to develop the subsystems of the end products that require further development.

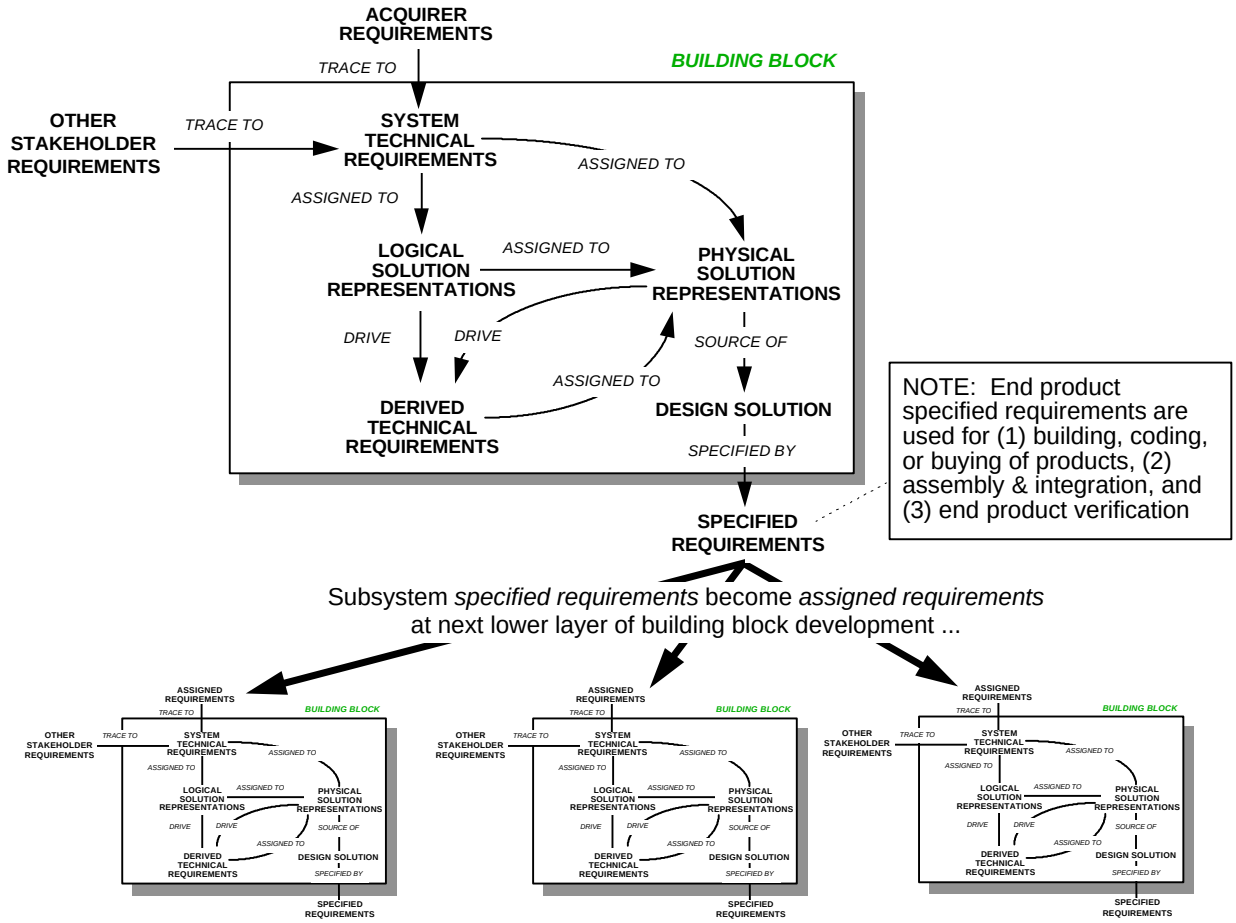

Figure G.2—Role of specified requirements

Enabling product requirements are generated during the application of the Requirement Definition and Solution Definition processes to the system, end product, and subsystem elements of the building block (see Subclause 6.1). These requirements are not shown in Figure G.1 since they become the basis for another building block development that uses these requirements as assigned requirements. An example of the development of two of the seven associated process enabling products is shown in Figure G.3.

The relationships between the types of requirements and the processes and process requirements of this Standard are shown in Figure G.4.

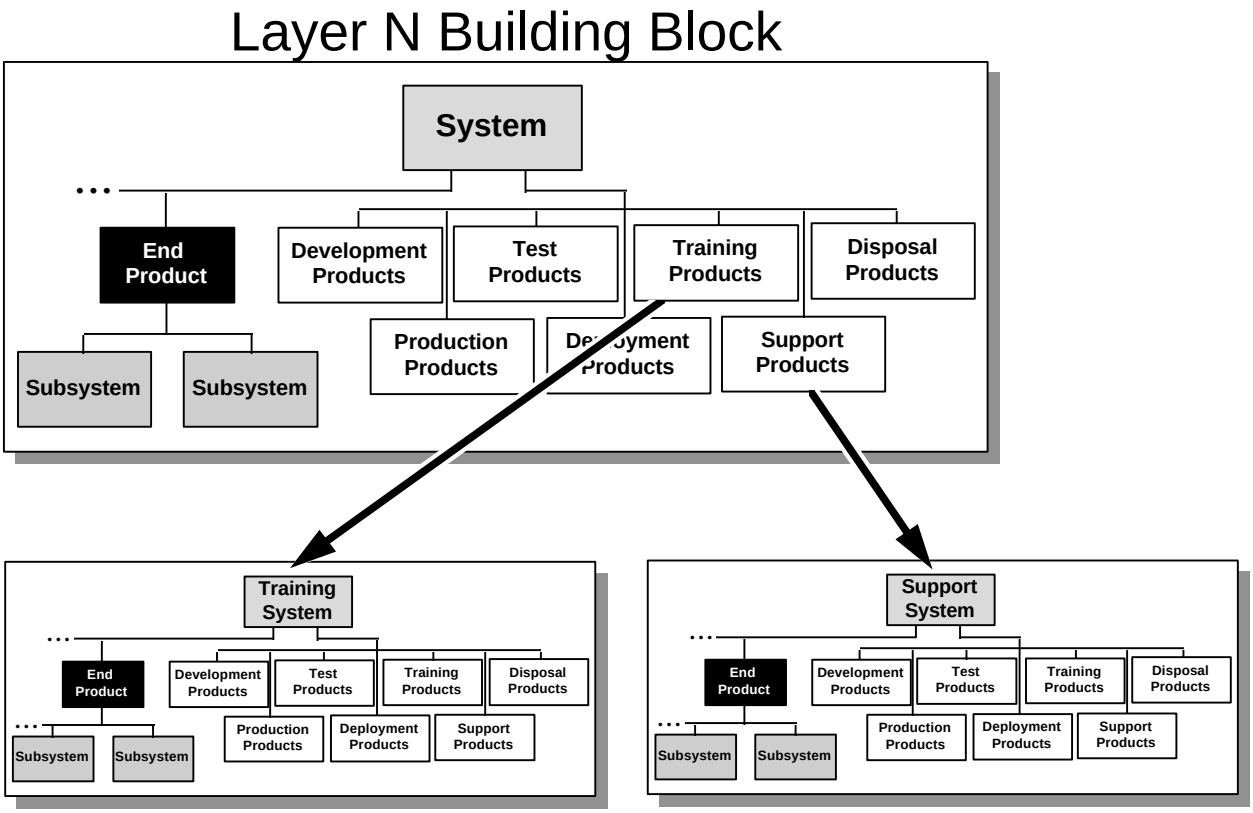

Figure G.3—Development of enabling products

Index

—F—

—H—

—I—

—L—

—M—

—N—

—O—

—P—

—Q—

—R—

—S—