

AS/NZS ISO/IEC/IEEE 15288:2023
ISO/IEC/IEEE 15288:2023



Australian/New Zealand Standard™

Systems and software engineering — System life cycle processes



AS/NZS ISO/IEC/IEEE 15288:2023

This Joint Australian/New Zealand Standard™ was prepared by Joint Technical Committee IT-015, Software and Systems Engineering. It was approved on behalf of the Council of Standards Australia on 29 September 2023 and by the New Zealand Standards Approval Board on 04 October 2023.

This Standard was published on 03 November 2023.

The following are represented on Committee IT-015:

- Australian Computer Society
- Australian Digital Health Agency
- Australian Society for Technical Communication (NSW)
- Business NSW
- Department of Defence (Australian Government)
- Engineers Australia
- Griffith University
- Institute of IT Professionals New Zealand
- IT Service Management Forum Australia
- Universities New Zealand
- University of New South Wales
- University of Southern Queensland
- University of Technology Sydney

This Standard was issued in draft form for comment as DR AS/NZS ISO/IEC/IEEE 15288:2023.

Keeping Standards up-to-date

Ensure you have the latest versions of our publications and keep up-to-date about Amendments, Rulings, Withdrawals, and new projects by visiting:

www.standards.org.au

www.standards.govt.nz

ISBN 978 1 76139 393 8

Australian/New Zealand Standard™

Systems and software engineering — System life cycle processes

Originated as AS/NZS 15288:2003.
Revised and designated as AS/NZS ISO/IEC/IEEE 15288:2015.
Second edition 2023.

COPYRIGHT

© ISO/IEC/IEEE 2023 — All rights reserved
© Standards Australia Limited/the Crown in right of New Zealand, administered by the New Zealand Standards Executive 2023

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher, unless otherwise permitted under the Copyright Act 1968 (Cth) or the Copyright Act 1994 (New Zealand).

Preface

This Standard was prepared by the Joint Standards Australia/Standards New Zealand Committee IT-015, Software and System Engineering, to supersede AS/NZS ISO/IEC/IEEE 15288:2015.

The objective of this document is to establish a common framework of process descriptions for describing the life cycle of systems created by humans, defining a set of processes and associated terminology from an engineering viewpoint.

This document applies to the full life cycle of systems, including —

- (a) conception;
- (b) development;
- (c) production;
- (d) utilization;
- (e) support; and
- (f) retirement of systems.

This document applies to various system types, including —

- (i) one-of-a-kind systems;
- (ii) mass-produced systems;
- (iii) customised and adaptable systems;
- (iv) a complete stand-alone system; and
- (v) to systems that are embedded and integrated into larger more complex and complete systems.

This document does not prescribe a specific system life cycle model, development methodology, method, modelling approach or technique.

This document does not detail information items in terms of name, format, explicit content, and recording media.

This document is identical with, and has been reproduced from, ISO/IEC IEEE 15288:2023, *Systems and software engineering — System life cycle processes*.

As this document has been reproduced from an International document, a full point substitutes for a comma when referring to a decimal marker.

Australian or Australian/New Zealand Standards that are identical adoptions of international normative references may be used interchangeably. Refer to the online catalogue for information on specific Standards.

The terms “normative” and “informative” are used in Standards to define the application of the appendices or annexes to which they apply. A “normative” appendix or annex is an integral part of a Standard, whereas an “informative” appendix or annex is only for information and guidance.

Contents

Preface	ii
Foreword	v
Introduction	vii
1 Scope	1
2 Normative references	1
3 Terms, definitions, and abbreviated terms	1
4 Conformance	9
4.1 Intended usage	9
4.2 Full conformance	10
4.2.1 Full conformance to outcomes	10
4.2.2 Full conformance to tasks	10
4.3 Tailored conformance	10
5 Key concepts and their application	11
5.1 General	11
5.2 System concepts	11
5.2.1 Systems	11
5.2.2 System structure	12
5.2.3 Interfacing, enabling, and interoperating systems	13
5.2.4 Concepts related to the system solution context	13
5.2.5 Product line engineering (PLE)	14
5.3 Organizational concepts	15
5.3.1 Organizations	15
5.3.2 Organization and project-level adoption	16
5.3.3 Organization and collaborative activities	16
5.4 System of systems concepts	16
5.4.1 Differences between systems and SoS	16
5.4.2 Managerial and operational independence	17
5.4.3 Taxonomy of SoS	17
5.4.4 SoS considerations in life cycle stages of a system	17
5.4.5 Application of this document to SoS	18
5.5 Life cycle concepts	18
5.5.1 System life cycle model	18
5.5.2 System life cycle stages	18
5.6 Process concepts	19
5.6.1 Criteria for processes	19
5.6.2 Description of processes	19
5.6.3 General characteristics of processes	19
5.7 Processes in this document	20
5.7.1 General	20
5.7.2 Agreement processes	22
5.7.3 Organizational project-enabling processes	22
5.7.4 Technical management processes	23
5.7.5 Technical processes	24
5.8 Process application	25
5.8.1 Overview	25
5.8.2 Process iteration, recursion, and concurrency	27
5.8.3 Process views	27
5.9 Concept and system definition	28
5.10 Assurance and quality characteristics	29
5.11 Process reference model	30
6 System life cycle processes	30
6.1 Agreement processes	30

6.1.1	Acquisition process.....	30
6.1.2	Supply process.....	32
6.2	Organizational project-enabling processes.....	34
6.2.1	Life cycle model management process.....	34
6.2.2	Infrastructure management process.....	36
6.2.3	Portfolio management process.....	37
6.2.4	Human resource management process.....	38
6.2.5	Quality management process.....	40
6.2.6	Knowledge management process.....	41
6.3	Technical management processes.....	43
6.3.1	Project planning process.....	43
6.3.2	Project assessment and control process.....	45
6.3.3	Decision management process.....	48
6.3.4	Risk management process.....	49
6.3.5	Configuration management process.....	52
6.3.6	Information management process.....	55
6.3.7	Measurement process.....	57
6.3.8	Quality assurance process.....	58
6.4	Technical processes.....	60
6.4.1	Business or mission analysis process.....	60
6.4.2	Stakeholder needs and requirements definition process.....	63
6.4.3	System requirements definition process.....	68
6.4.4	System architecture definition process.....	71
6.4.5	Design definition process.....	74
6.4.6	System analysis process.....	77
6.4.7	Implementation process.....	79
6.4.8	Integration process.....	82
6.4.9	Verification process.....	84
6.4.10	Transition process.....	87
6.4.11	Validation process.....	90
6.4.12	Operation process.....	93
6.4.13	Maintenance process.....	96
6.4.14	Disposal process.....	100
Annex A	(normative) Tailoring process.....	103
Annex B	(informative) Example process artefacts and information items.....	105
Annex C	(informative) Process reference model for assessment purposes.....	109
Annex D	(informative) Model-based systems and software engineering (MBSSE).....	111
Bibliography		115
IEEE notices and abstract		119

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialised system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO/IEC documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see <https://patents.iec.ch>).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*, in cooperation with the Systems and Software Engineering Standards Committee of the IEEE Computer Society, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

This second edition cancels and replaces the first edition (ISO/IEC/IEEE 15288:2015), which has been technically revised.

The main changes are as follows:

- improvements to selected technical processes including business or mission analysis, system architecture definition, system analysis, implementation, integration, operations, and maintenance;
- improvements to selected technical management processes including risk management and configuration management;
- updates to [Clause 5](#), key concepts, including a better description of iteration, recursion, system-of-systems, quality characteristics, etc.;

- new content in [Clause 5](#) on concept and system definition, and expanded content on process application and system concepts;
- updates to the terms and definitions;
- a new annex addressing model-based systems engineering (MBSE).

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

The complexity of systems continues to increase to unprecedented levels. This has led to new opportunities, but also to increased challenges for the organizations that create and utilise systems. These challenges exist throughout the life cycle of a system and at all levels of architectural detail. This document provides a common process framework for describing the life cycle of systems, adopting a systems engineering approach. This document concerns systems that can be configured with one or more of the following system elements: hardware elements, software elements, data, humans, processes, services, procedures, facilities, materials, and naturally occurring entities.

This document focuses on defining stakeholder needs, concerns, priorities, and constraints for the required functionality early in the development cycle, establishing requirements, then proceeding with design synthesis and system validation while considering the complete problem. It integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from conception through production to operation. It considers the needs of all stakeholders with the goal of providing a quality product that meets the needs of users and other applicable stakeholders. It provides the processes for acquiring and supplying systems. It helps to improve communication and cooperation among the parties that create, utilise, and manage modern systems in order that they can work in an integrated, coherent fashion. Finally, this document provides the framework for assessment and improvement of the life cycle processes.

There is a wide variety of systems in terms of their purpose, domain of application, complexity, size, novelty, adaptability, quantity, location, life span, and evolution. The processes in this document form a comprehensive set from which an organization can construct system life cycle models appropriate to its products and services. An organization, depending on its purpose, can select and apply an appropriate subset to fulfil that purpose.

This document can be used in one or more of the following modes:

- By an organization — to help establish an environment of desired processes. These processes can be supported by an infrastructure of methods, procedures, techniques, tools, and trained personnel. The organization may then employ this environment to perform and manage its projects and progress systems through their life cycle stages. In this mode this document is used to assess conformance of a declared, established environment to its provisions. It can be used by a single organization in a self-imposed mode or in a multi-party situation. Parties can be from the same organization or from different organizations and the situation can range from an informal agreement to a formal contract.
- By a project — to help select, structure, and employ the elements of an established environment to provide products and services. In this mode this document is used in the assessment of conformance of the project to the declared and established environment.
- By an acquirer and a supplier — to help develop an agreement concerning processes and activities. Via the agreement, the processes and activities in this document are selected, negotiated, agreed to, and performed. In this mode this document is used for guidance in developing the agreement.
- By process assessors — to serve as a process reference model for use in the performance of process assessments that can be used to support organizational process improvement.

In the context of this document and ISO/IEC/IEEE 12207, there is a continuum of human-made systems from those that use little or no software to those in which software is the primary interest. When software is the predominant system or element of interest, ISO/IEC/IEEE 12207 should be used. Both documents have the same process model, share most activities and tasks, and differ primarily in descriptive notes.

Although this document does not establish a management system, it is intended to be compatible with the quality management system provided by ISO 9001, the service management system provided by

ISO/IEC 20000 series, the IT asset management system provided by the ISO/IEC 19770 series, and the information security management system provided by ISO/IEC 27000.

Australian/New Zealand Standard

Systems and software engineering — System life cycle processes

1 Scope

This document establishes a common framework of process descriptions for describing the life cycle of systems created by humans, defining a set of processes and associated terminology from an engineering viewpoint. These processes can be applied to systems of interest, their system elements, and to systems of systems. Selected sets of these processes can be applied throughout the stages of a system's life cycle. This is accomplished through the involvement of stakeholders, with the ultimate goal of achieving customer satisfaction.

This document defines a set of processes to facilitate system development and information exchange among acquirers, suppliers, and other stakeholders in the life cycle of a system.

This document specifies processes that support the definition, control, and improvement of the system life cycle processes used within an organization or a project. Organizations and projects can use these processes when acquiring and supplying systems.

This document applies to organizations in their roles as both acquirers and suppliers.

This document applies to the full life cycle of systems, including conception, development, production, utilization, support and retirement of systems, and to the acquisition and supply of systems, whether performed internally or externally to an organization. The life cycle processes of this document can be applied iteratively and concurrently to a system and recursively to the system elements.

This document applies to one-of-a-kind systems, mass-produced systems, and customised, adaptable systems. It also applies to a complete stand-alone system and to systems that are embedded and integrated into larger more complex and complete systems.

This document does not prescribe a specific system life cycle model, development methodology, method, modelling approach or technique.

This document does not detail information items in terms of name, format, explicit content, and recording media. ISO/IEC/IEEE 15289 addresses the content for life cycle process information items (documentation).

2 Normative references

There are no normative references in this document.

3 Terms, definitions, and abbreviated terms

For the purposes of this document, the following terms and definitions apply.

ISO, IEC, and IEEE maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>
- IEEE Standards Dictionary Online: available at: <https://dictionary.ieee.org/>

NOTE Definitions for other system and software engineering terms can be found in ISO/IEC/IEEE 24765, available at www.computer.org/sevocab.

3.1 acquirer

stakeholder (3.44) that acquires or procures a *system* (3.46), *product* (3.32) or *service* (3.42) from a *supplier* (3.45)

Note 1 to entry: Other terms commonly used for an acquirer are buyer, *customer* (3.12), owner, purchaser, or internal/organizational sponsor.

3.2 acquisition

process (3.27) of obtaining a *system* (3.46), *product* (3.32) or *service* (3.42)

3.3 activity

set of cohesive *tasks* (3.51) of a *process* (3.27)

3.4 agreement

mutual acknowledgement of terms and conditions under which a working relationship is conducted

EXAMPLE Contract, memorandum of agreement.

3.5 architecture

fundamental concepts or properties of a *system* (3.46) in its *environment* (3.16) and governing principles for the realization and evolution of this system and its related *life cycle* (3.21) *processes* (3.27)

[SOURCE: ISO/IEC/IEEE 42020:2019, 3.3, modified — 'entity' has been replaced with 'system'; notes to entry have been removed.]

3.6 artefact

work *product* (3.32) that is produced and used during a project to capture and convey information

EXAMPLE Models, stakeholder requirements, system/software requirements, architecture descriptions, design descriptions, source code, implemented system elements, verified or validated system.

[SOURCE: ISO 19014-4:2020, 3.9, modified — EXAMPLE has been added.]

3.7 audit

independent examination of a work *product* (3.32) or set of work products to assess compliance with specifications, standards, contractual *agreements* (3.4), or other criteria

3.8 baseline

formally approved version of a *configuration item* (3.11), regardless of media, formally designated and fixed at a specific time during the configuration item's *life cycle* (3.21)

[SOURCE: IEEE Std 828-2012]

3.9 concept of operations

verbal and graphic statement, in broad outline, of an *organization's* (3.25) assumptions or intent in regard to an operation or series of operations of new, modified, or existing organizational *systems* (3.46)

Note 1 to entry: The concept of operations frequently is embodied in long-range strategic plans and annual operational plans. In the latter case, the concept of operations in the plan covers a series of connected operations to be carried out simultaneously or in succession to achieve an organizational performance objective. See also *operational concept* (3.23).

Note 2 to entry: The concept of operations provides the basis for bounding the operating space, system capabilities, *interfaces* (3.19), and operating *environment* (3.16).

[SOURCE: ANSI/AIAA G-043B-2018, 5.2, modified — The second definition has been used; the last two sentences of Note 1 to entry have been removed; Note 2 to entry has been added.]

3.10

concern

matter of interest or importance to a *stakeholder* (3.44)

Note 1 to entry: A concern pertains to any influence on a *system* (3.46) in its *environment* (3.16), including developmental, technological, business, operational, organizational, political, economic, legal, regulatory, ethical, ecological, and social influences.

[SOURCE: ISO/IEC/IEEE 42020:2019, 3.8, modified — EXAMPLE has been removed; Note 1 to entry has been added.]

3.11

configuration item

item or aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management *process* (3.27)

3.12

customer

organization (3.25) or person that receives a *product* (3.32) or *service* (3.42)

EXAMPLE Consumer, client, *user* (3.53), *acquirer* (3.1), buyer, or purchaser.

Note 1 to entry: A customer can be internal or external to the organization.

3.13

design, noun

specification of *system elements* (3.47) and their relationships, that is sufficiently complete to support a compliant implementation of the *architecture* (3.5)

Note 1 to entry: Design provides the detailed implementation-level physical structure, behaviour, temporal relationships, and other attributes of system elements.

3.14

design characteristics

design attributes or distinguishing features that pertain to a measurable description of a *product* (3.32) or *service* (3.42)

3.15

enabling system

system (3.46) that supports a *system-of-interest* (3.48) during its *life cycle* (3.21) *stages* (3.43) but does not necessarily contribute directly to its function during operation

EXAMPLE Production-enabling system, which is required when a system-of-interest enters the production stage.

Note 1 to entry: Each enabling system has a life cycle of its own. This document is applicable to each enabling system when, in its own right, it is treated as a system-of-interest.

3.16

environment

<system> context determining the setting and circumstances of all influences upon a *system* (3.46)

3.17

incident

anomalous or unexpected event, set of events, condition, or situation at any time during the *life cycle* (3.21) of a *project* (3.33), *product* (3.32), *service* (3.42), or *system* (3.46)

Note 1 to entry: An incident is elevated and treated as a *problem* (3.26) when the cause of the incident needs to be analysed and corrected to prevent reoccurrence to avoid or minimise loss of life, or damage of property or natural *resources* (3.37).

3.18 information item

separately identifiable body of information that is produced, stored, and delivered for human use

[SOURCE: ISO/IEC/IEEE 15289:2019, 3.1.12, modified — The preferred term "information product" has been removed; notes to entry have been removed.]

3.19 interface

point at which two or more logical, physical, or both, *system elements* (3.47) or software system elements meet and act on or communicate with each other

[SOURCE: ISO/IEC/IEEE 24748-6: —, 3.1.3]

3.20 interoperating system

system (3.46) that exchanges information with the *system-of-interest* (3.48) and uses the information that has been exchanged

3.21 life cycle

evolution of a *system* (3.46), *product* (3.32), *service* (3.42), *project* (3.33) or other human-made entity from conception through *retirement* (3.38)

3.22 life cycle model

framework of *processes* (3.27) and *activities* (3.3) concerned with the *life cycle* (3.21) which can be organized into *stages* (3.43), acting as a common reference for communication and understanding

3.23 operational concept

verbal and graphic statement of an *organization's* (3.25) assumptions or intent in regard to an operation or series of operations of a specific *system* (3.46) or a related set of specific new, existing or modified systems

Note 1 to entry: The operational concept is designed to give an overall picture of the operations using one or more specific systems, or set of related systems, in the organization's operational *environment* (3.16) from the *users'* (3.53) and *operators'* (3.24) perspectives. See also *concept of operations* (3.9).

Note 2 to entry: The operational concept is about systems, while a *concept of operations* (3.9) typically refers to organizations.

[SOURCE: ANSI/AIAA G-043B-2018, 5.2, modified — The third definition has been used; the first sentence in Note 1 to entry has been removed; Note 2 to entry has been added.]

3.24 operator

individual or *organization* (3.25) that performs the operations of a *system* (3.46)

Note 1 to entry: The role of operator and the role of *user* (3.53) can be vested, simultaneously or sequentially, in the same individual or organization.

Note 2 to entry: An individual operator combined with knowledge, skills, and procedures can be considered as an element of the system.

Note 3 to entry: An operator may perform operations on a system that is operated, or of a system that is operated, depending on whether or not operating instructions are placed within the system boundary.

3.25 organization

person or group of people that has its own functions with responsibilities, authorities, and relationships to achieve its objectives

EXAMPLE Company, corporation, firm, enterprise, manufacturer, institution, charity, sole trader, association, or parts or combination thereof.

[SOURCE: ISO 9000:2015, 3.2.1, modified — Notes to entry have been removed; EXAMPLE has been added.]

3.26

problem

difficulty, uncertainty, or otherwise realised and undesirable event, set of events, condition, or situation that requires investigation and corrective action

3.27

process

set of interrelated or interacting *activities* ([3.3](#)) that transform inputs into outputs

3.28

iteration

<process> repeating the application of the same *process* ([3.27](#)) or set of processes on the same level of the *system* ([3.46](#)) structure

3.29

process purpose

high level objective of performing the *process* ([3.27](#)) and the likely outcomes of effective implementation of the process

Note 1 to entry: The purpose of implementing the process is to provide benefits to the *stakeholders* ([3.44](#)).

3.30

process outcome

observable result of the successful achievement of the *process purpose* ([3.29](#))

3.31

recursion

<process> repeating the application of the same *process* ([3.27](#)) or set of processes to successive levels of *system elements* ([3.47](#)) in the system structure

3.32

product

output of an *organization* ([3.25](#)) that can be produced without any transaction taking place between the organization and the *customer* ([3.12](#))

Note 1 to entry: The dominant element of a product is that it is generally tangible.

[SOURCE: ISO 9000:2015, 3.7.6, modified — Notes 1 and 3 to entry have been removed.]

3.33

project

endeavour with defined start and finish criteria undertaken to create a *product* ([3.32](#)) or *service* ([3.42](#)) in accordance with specified *resources* ([3.37](#)) and *requirements* ([3.36](#))

Note 1 to entry: A project is sometimes viewed as a unique *process* ([3.27](#)) comprising coordinated and controlled *activities* ([3.3](#)) and composed of activities from the Technical Management and Technical Processes defined in this document.

Note 2 to entry: Continuous development approaches such as agile and DevOps can use different terminology for the creation of product and services.

3.34 quality assurance QA

part of quality management focused on providing confidence that quality *requirements* (3.36) will be fulfilled

[SOURCE: ISO 9000:2015, 3.3.6, modified — The abbreviated term has been added.]

3.35 quality characteristic

inherent characteristic of a *product* (3.32), *service* (3.42), *process* (3.27), or *system* (3.46) related to a *requirement* (3.36)

[SOURCE: ISO 9000:2015, 3.10.2, modified — ‘object’ has been replaced with ‘product, service, process, or system’; notes to entry have been removed.]

3.36 requirement

statement which translates or expresses a need and its associated constraints and conditions

[SOURCE: ISO/IEC/IEEE 29148:2018, 3.1.19, modified — Notes to entry have been removed.]

3.37 resource

asset that is utilised or consumed during the execution of a *process* (3.27)

Note 1 to entry: Resource includes diverse entities such as funding, personnel, facilities, capital equipment, tools and utilities such as power, water, fuel, and communication infrastructures.

Note 2 to entry: Resources include those that are reusable, renewable or consumable.

3.38 retirement

<system> withdrawal of active support by the operation and maintenance *organization* (3.25), partial or total replacement by a new *system* (3.46), or installation of an upgraded system, or final decommissioning and disposal

3.39 risk

effect of uncertainty on objectives

Note 1 to entry: An effect is a deviation from the expected — positive or negative. A positive effect is also known as an opportunity.

Note 2 to entry: Objectives can have different aspects [such as financial, health and *safety* (3.40), and environmental goals] and can apply at different levels [such as strategic, organization-wide, project, *product* (3.32) and *process* (3.27)].

Note 3 to entry: Risk is often characterized by reference to potential events and consequences, or a combination of these.

Note 4 to entry: Risk is often expressed in terms of a combination of the consequences of an event (including changes in circumstances) and the associated likelihood of occurrence.

Note 5 to entry: Uncertainty is the state, even partial, of deficiency of information related to understanding or knowledge of an event, its consequence, or likelihood.

[SOURCE: ISO Guide 73:2009, 1.1, modified — The last sentence in Note 1 to entry has been added.]

3.40 safety

expectation that a *system* (3.46) does not, under defined conditions, lead to a state in which human life, health, property, or the *environment* (3.16) is endangered

Note 1 to entry: The term is alternatively defined as freedom from *risk* (3.39) which is not tolerable.

[SOURCE: ISO/IEC/IEEE 12207:2017, 3.1.48, modified — Note 1 to entry has been added.]

3.41 security

protection against intentional subversion or forced failure

Note 1 to entry: Security includes authenticity, accountability, confidentiality, integrity, availability, non-repudiation, and reliability, all of which have the related issue of their assurance.

[SOURCE: NATO AEP-67, modified — Note 1 to entry has been updated.]

3.42 service

output of an *organization* (3.25) with at least one *activity* (3.3) necessarily performed between the organization and the *customer* (3.12)

Note 1 to entry: The dominant elements of a service are generally intangible.

Note 2 to entry: A service is coherent, discrete, and can be composed of other services.

[SOURCE: ISO 9000:2015, 3.7.7, modified — Notes 2, 3, and 4 to entry have been replaced by a new Note 2 to entry.]

3.43 stage

period within the *life cycle* (3.21) of an entity that relates to the state of its description or realization

Note 1 to entry: As used in this document, stages relate to major progress and achievement milestones of the entity through its life cycle.

Note 2 to entry: Stages often overlap.

3.44 stakeholder

individual or *organization* (3.25) having a right, share, claim, or interest in a *system* (3.46) or in its possession of characteristics that meet their needs and expectations

EXAMPLE End users (3.53), end user organizations, supporters, developers, customers (3.12), producers, trainers, maintainers, disposers, acquirers (3.1), suppliers (3.45), regulatory bodies, and people influenced positively or negatively by a system.

Note 1 to entry: Some stakeholders can have interests that oppose each other or oppose the system.

3.45 supplier

organization (3.25) or an individual that enters into an *agreement* (3.4) with the *acquirer* (3.1) for the supply of a *product* (3.32) or *service* (3.42)

Note 1 to entry: Other terms commonly used for supplier are contractor, producer, seller or vendor.

Note 2 to entry: The acquirer and the supplier sometimes are part of the same organization.

3.46 system

arrangement of parts or elements that together exhibit a stated behaviour or meaning that the individual constituents do not

Note 1 to entry: A system is sometimes considered as a *product* (3.32) or as the *services* (3.42) it provides.

Note 2 to entry: In practice, the interpretation of its meaning is frequently clarified by the use of an associative noun, e.g. aircraft system. Alternatively, the word "system" is substituted simply by a context-dependent synonym (e.g. aircraft), though this potentially obscures a system principles perspective.

Note 3 to entry: A complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, *services* (3.42), and personnel required for operations and support to the degree necessary for self-sufficient use in its intended *environment* (3.16).

3.47 system element

discrete part of a *system* (3.46) that can be implemented to fulfil specified *requirements* (3.36)

EXAMPLE Hardware, software, data, humans, *processes* (3.27) [e.g. processes for providing *service* (3.42) to *users* (3.53)], procedures [e.g., *operator* (3.24) instructions], facilities, materials, and naturally occurring entities or any combination.

3.48 system-of-interest

SoI
system (3.46) whose *life cycle* (3.21) is under consideration

3.49 system of systems

SoS
set of *systems* (3.46) or *system elements* (3.47) that interact to provide a unique capability that none of the constituent systems can accomplish on its own

[SOURCE: ISO/IEC/IEEE 21839:2019, 3.1.4]

3.50 systems engineering

transdisciplinary and integrative approach to enable the successful realization, use, and *retirement* (3.38) of engineered *systems* (3.46) using systems principles and concepts and scientific, technological and management methods

[SOURCE: INCOSE-TP-2020-002-06]

3.51 task

required, recommended, or permissible action, intended to contribute to the achievement of one or more outcomes of a *process* (3.27)

3.52 traceability

discernible association among two or more logical entities, such as *requirements* (3.36), *system elements* (3.47), *verifications* (3.55), or *tasks* (3.51)

[SOURCE: ISO/IEC TR 29110-1:2016, 3.71, modified — "discernible" has been added; EXAMPLE has been removed.]

3.53**user**

individual or group that interacts with a *system* (3.46) or benefits from a system during its utilization

Note 1 to entry: The role of user and the role of *operator* (3.24) are sometimes vested, simultaneously or sequentially, in the same individual or *organization* (3.25).

[SOURCE: ISO/IEC 25010:2011, 4.3.16, modified — The original Note 1 to entry has been replaced by a new one.]

3.54**validation**

confirmation, through the provision of objective evidence, that the *requirements* (3.36) for a specific intended use or application have been fulfilled

Note 1 to entry: In a *life cycle* (3.21) context, validation involves the set of *activities* (3.3) for gaining confidence that a *system* (3.46) is able to accomplish its intended use, goals, and objectives in an *environment* (3.16) like the operational environment. The right system was built.

[SOURCE: ISO 9000:2015, 3.8.13, modified — Notes 1 to 3 to entry have been removed; a new Note 1 to entry has been added.]

3.55**verification**

confirmation, through the provision of objective evidence, that specified *requirements* (3.36) have been fulfilled

Note 1 to entry: Verification is a set of *activities* (3.3) that compares a *system* (3.46) or *system element* (3.47) against the required characteristics. This includes, but is not limited to, specified requirements, *design* (3.13) description, and the system itself. The system was built right.

[SOURCE: ISO 9000:2015, 3.8.12, modified — Notes 1 to 3 to entry have been removed; a new Note 1 to entry has been added.]

3.56**view**

representation of a *system* (3.46) from the perspective of a related set of *concerns* (3.10)

Note 1 to entry: A view can be an operational, functional, or architectural representation of a system.

[SOURCE: ISO/IEC/IEEE 24774:2021, 3.21, modified — removed ‘whole’ from the definition and the original Note 1 to entry has been replaced by a new one.]

3.57**viewpoint**

specification of the conventions for constructing and using a *view* (3.56)

[SOURCE: ISO/IEC/IEEE 24774:2021, 3.22, modified — Notes 1 to 3 to entry have been removed.]

4 Conformance**4.1 Intended usage**

The requirements in this document are contained in [Clause 6](#) and [Annex A](#). This document provides requirements for a number of processes suitable for usage during the life cycle of a system. It is possible that particular projects or organizations need only some of the processes provided by this document. Therefore, implementation of this document typically involves selecting and declaring a set of processes suitable to the organization or project. There are two ways that an implementation can be claimed to conform to the provisions of this document – full conformance and tailored conformance.

There are two criteria for claiming full conformance. Achieving either criterion suffices for conformance, although the chosen criterion (or criteria) shall be stated in the claim. Claiming “full conformance to tasks” asserts that all of the requirements of the activities and tasks of the declared set of processes are achieved. Alternatively, claiming “full conformance to outcomes” asserts that all of the required outcomes of the declared set of processes are achieved. Full conformance to outcomes permits greater freedom in the implementation of conforming processes and can be useful for implementing processes to be used in the context of an innovative life cycle model.

NOTE 1 Options for conformance are provided for needed flexibility in the application of this document. Each process has a set of objectives (phrased as “outcomes”) and a set of activities and tasks that represent one way to achieve the objectives.

NOTE 2 Users who implement the activities and tasks of the declared set of processes can assert full conformance to tasks of the selected processes. Some users, however, can have innovative process variants that achieve the objectives (i.e. the outcomes) of the declared set of processes without implementing all of the activities and tasks. These users can assert full conformance to the outcomes of the declared set of processes. The two criteria – conformance to task and conformance to outcome – are necessarily not equivalent since specific performance of activities and tasks can require, in some cases, a higher level of capability than just the achievement of outcomes.

NOTE 3 An organization (e.g. national, industrial association, company) imposing this document as a condition of trade can specify and make public the minimum set of required processes, outcomes, activities, and tasks, which constitute suppliers' compliance with the conditions of trade.

NOTE 4 Requirements of this document are marked by the use of the verb “shall”. Recommendations are marked by the use of the verb “should”. Permissions are marked by the use of the verb “may”. However, despite the verb that is used, the requirements for conformance are selected as described previously.

4.2 Full conformance

4.2.1 Full conformance to outcomes

A claim of full conformance declares the set of processes for which conformance is claimed. Full conformance to outcomes is achieved by demonstrating that all of the outcomes of the declared set of processes have been achieved. In this situation, the provisions for activities and tasks of the declared set of processes are guidance rather than requirements, regardless of the verb form that is used in the provision.

NOTE One intended use of this document is to facilitate process assessment and improvement. For this purpose, the objectives of each process are written in the form of ‘outcomes’ compatible with the provisions of the ISO/IEC 33000 family of standards. Those standards provide for the assessment of the processes of this document, providing a basis for improvement. Users intending process assessment and improvement can use the process outcomes written in this document as the “process reference model” required by ISO/IEC 33002.

4.2.2 Full conformance to tasks

A claim of full conformance declares the set of processes for which conformance is claimed. Full conformance to tasks is achieved by demonstrating that all of the requirements of the activities and tasks of the declared set of processes have been achieved. In this situation, the provisions for the outcomes of the declared set of processes are guidance rather than requirements, regardless of the verb form that is used in the provision.

NOTE A claim of full conformance to tasks can be appropriate in contractual situations where an acquirer or a regulator requires detailed understanding of the suppliers’ processes.

4.3 Tailored conformance

When this document is used as a basis for establishing a set of processes that do not qualify for full conformance, the processes in [Clause 6](#) of this document shall be selected or modified in accordance with the tailoring process prescribed in [Annex A](#). The tailored set of processes, for which tailored

conformance is claimed, are declared. Tailored conformance is achieved by demonstrating that the outcomes, activities, and tasks, as tailored, have been achieved.

NOTE 1 Tailoring can diminish the perceived value of a claim of conformance to this document because it is difficult for other organizations to understand the extent to which tailoring can have deleted desirable provisions.

NOTE 2 An organization asserting a claim of conformance to this document can find it advantageous to claim full conformance to a smaller list of processes rather than tailored conformance to a larger list of processes.

NOTE 3 An organization can also choose to claim full conformance to a selected set of processes as well as tailored conformance to other processes.

5 Key concepts and their application

5.1 General

This clause highlights and explains essential concepts on which this document is based. Further elaboration of these concepts can be found in the ISO/IEC/IEEE 24748-1 and ISO/IEC/IEEE 24748-2, which provide guidelines on the application of life cycle management.

5.2 System concepts

5.2.1 Systems

A system is an arrangement of parts or elements that together exhibit behaviour or meaning that the individual constituents do not. Systems can be either physical or conceptual, or a combination of both. Systems in the physical universe are composed of matter and energy, may embody information encoded in matter-energy carriers, and exhibit observable behaviour. Conceptual systems are abstract systems of pure information, and do not directly exhibit behaviour, but exhibit “meaning”. In both cases, the system’s properties (as a whole) result or emerge from the parts or elements and their individual properties and the relationships and interactions between and among the parts, the system and its environment.

The perception and definition of a particular system, its architecture and its system elements depend on a stakeholder’s interests and responsibilities. One stakeholder’s system-of-interest (SoI) can be viewed as a system element in another stakeholder’s SoI. Furthermore, an SoI can be viewed as being part of the environment for another stakeholder’s SoI. Also, an SoI can be viewed as a constituent system in an SoS.

The following are key points regarding the characteristics of the SoI (also see [Figure 1](#) and [Figure 2](#)):

- a) defined boundaries encapsulate meaningful needs and practical solutions;
- b) there is a hierarchical or other relationship between system elements;
- c) an entity at any level in the SoI can be viewed as a system;
- d) a system comprises an integrated, defined set of subordinate system elements;
- e) humans can be viewed as both users external to a system (e.g. users) and as system elements (e.g. operators) within a system;
- f) a system can be viewed in isolation as an entity, i.e. a product; or as a collection of functions capable of interacting with its surrounding environment, i.e. a set of services.

NOTE Services, hardware elements, and software elements can be products or services if they are either individual SoIs or constituents in an SoS. Consideration of services and products integrated within a system is sometimes called a product-service system^[65]. Product-service systems provide a means for organizations to offer services related to their products.

Whatever the boundaries chosen to define the system, the concepts in this document are generic and permit a practitioner to correlate or adapt individual instances of life cycles to its system principles.

5.2.2 System structure

The system life cycle processes in this document are described in relation to a system (see [Figure 1](#)) which is composed of a set of interacting system elements, each of which can be implemented to fulfil its respective specified requirements. System elements may include software elements, hardware elements, services, and utilization and support resources. Responsibility for the implementation of any system element may be delegated to another party through an agreement.

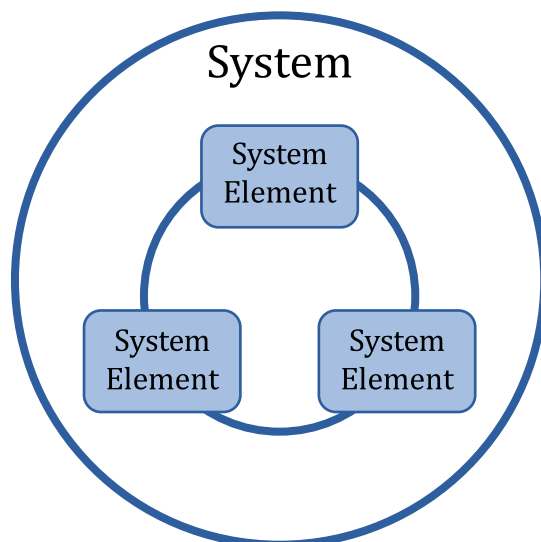


Figure 1 — System and system element relationship

The relationship between system elements can be expressed in many forms, including hierarchies or networks. For more complex SoIs, a prospective system element may itself need to be considered as a system (that in turn is comprised of system elements) before a complete set of system elements can be defined with confidence (see [Figure 2](#)). In this manner, the appropriate system life cycle processes are applied recursively to an SoI to resolve its structure to the point where understandable and manageable system elements can be implemented (made, bought, or reused). While [Figures 1](#) and [2](#) imply a hierarchical relationship, in reality there are an increasing number of systems that, from one or more aspects, are not hierarchical, such as networks and other distributed systems. [5.4](#) discusses the concept of a system of systems (SoS).

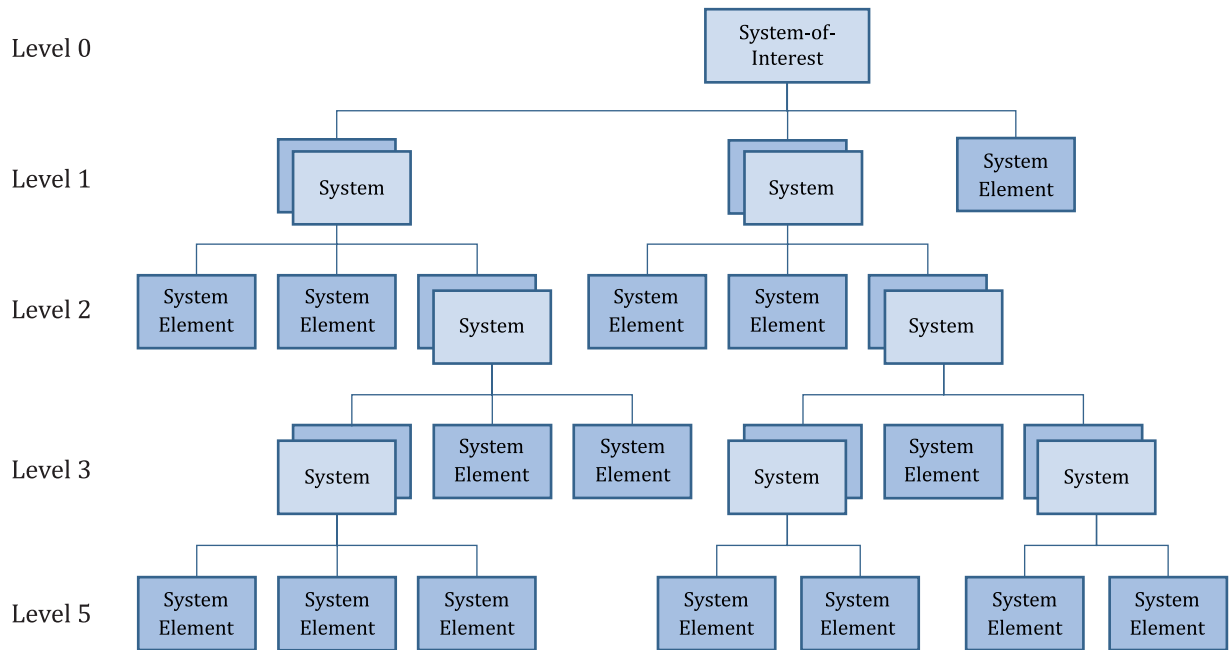


Figure 2 — System-of-interest structure

5.2.3 Interfacing, enabling, and interoperating systems

Any system sharing an interface (data or information, energy, resource, physical) with the Sol during any stage of the Sol's life cycle is an interfacing system and needs to be considered in the system development. Humans can be system elements of the Sol (e.g. an operator) or can be interfacing externally to the Sol (e.g. a user requesting information) throughout the Sol's life cycle stages.

Throughout the life cycle of an Sol, essential services are required from enabling systems, e.g. mass-production system, training system, maintenance system. Each of these systems supports one or more lifecycle processes of the Sol to be conducted. Sol's often have interfaces with other systems that are used during life cycle stages other than operations. Some of these interfaces can be exclusive to that stage and not used during operation.

Systems that interact to perform a function are called interoperating systems, which are an important aspect in the context of systems of systems (see 5.4). Interoperating systems are a subset of the interfacing systems. While interoperability can involve the exchange and use of information, physical and other types of interoperability can be important. For example, many kinds of electronic devices now have power adapters appropriate for the user's location.

NOTE Interoperating systems can exchange information to enable an Sol to operate reliably, securely, usefully or efficiently, or to improve accessibility or usability. An interoperating system can also receive information from the Sol for use by other systems or SoS.

The interrelationships between the Sol and the interfacing, enabling, and interoperating systems can be bi-directional or one-way. Requirements for the interrelationships need to be included in the requirements for the Sol.

Further elaboration of these concepts can be found in the ISO/IEC/IEEE 24748-1 and ISO/IEC/IEEE 24748-2, which provide guidelines on the application of life cycle processes.

5.2.4 Concepts related to the system solution context

An Sol and its enabling systems are normally thought of as a solution addressing stakeholder concerns. The concerns of stakeholders are related to their business models. In particular, the concerns of an Sol supplier, an acquirer, and a user are different; this drives the need for them to consider different

enabling systems to make the SoI viable in their own system solution context. Thus, the solution needs to consider the different stakeholder needs and business models.

EXAMPLE A manufacturing system is necessary for the supplier and is usually not considered by users and acquirers.

In this perspective, for a given SoI, the business or mission analysis process is intended to address the set of solution contexts (see [Figure 3](#)).

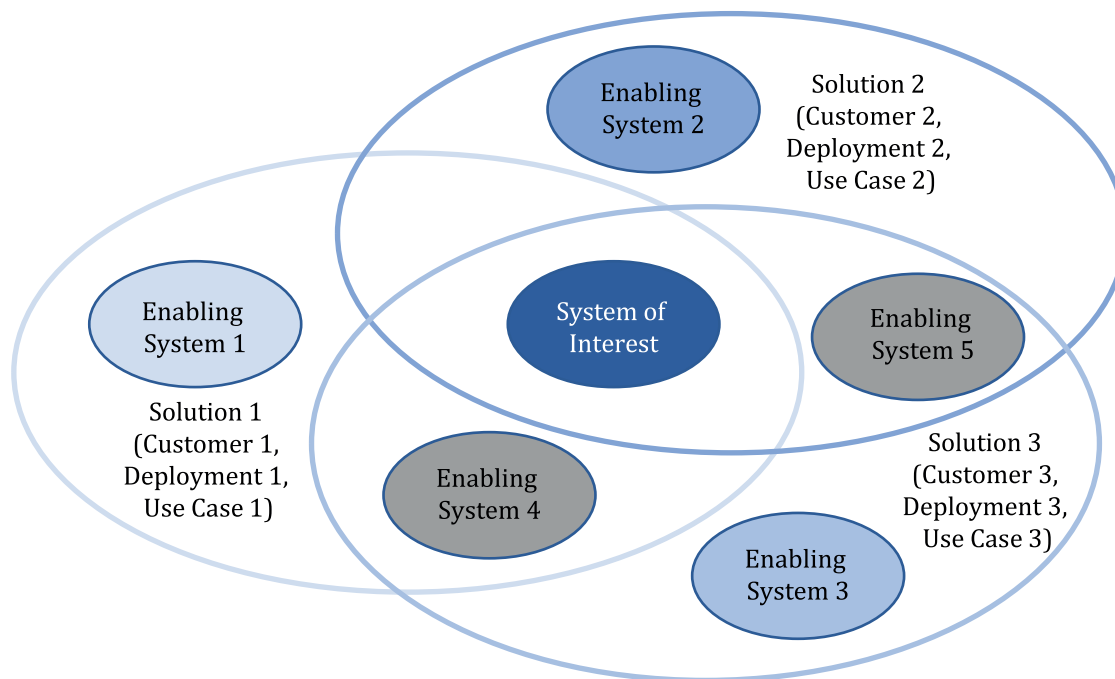


Figure 3 — System solution contexts

NOTE Several operational concepts, acquisition contexts and deployments can be associated with a given SoI. This multi-dimensional life description is provided in ISO 15704.

Variants and options shall be specified for the SoI to address the set of solution contexts. Products and services are often considered in system families and product lines with identification of elements common to different projects, and variants and options per project (see ISO/IEC 26550 for more details). Development of systems, products, and services often benefit from the identification of reuse opportunities between projects, including the establishment of product lines, families of products, systems, and systems of systems. These assets available for the projects are managed through the application of the knowledge management process.

5.2.5 Product line engineering (PLE)

When an organization develops a product line, engineering the product line holistically is much more effective and efficient than engineering each of the individual systems. This requires engineering the product line as a single SoI, with variations defined to support the individual system instances, for much of the life cycle. However, at the point in the life cycle where a specific system instance is developed, validated, and deployed into operation – an individual member of the product line – that system becomes an SoI itself that can continue with its own post-development life cycle, which can include production, support, utilization, retirement, and more. Variation management models are applied to manage the definition and production of the system instances. Whereas PLE generally focuses on the benefits of using a common platform with reusable assets for a product family, feature-based PLE addresses PLE in a holistic and automated manner (see ISO/IEC 26580).

In this approach, all of the system life cycle processes apply both when the product line is viewed as the SoI and when each instance of the product line with its variations is considered the SoI. From the

holistic product line SoI perspective, many of the artefacts developed through the life cycle processes are shared across multiple members of the product line, which adds to the efficiency.

The following are key tenets of feature-based PLE:

- A collective set of features for the system instances in the product line (called the feature catalogue in ISO/IEC 26580) captures the distinguishing characteristics of how the members of the product line differ from each other and provides a common language of variation throughout the organization. The feature catalogue is a special type of MBSE (model-based systems engineering) model that helps to analyse and address the variations in the product line (see [Annex D](#)).
- The features selected for a system instance in a product line portfolio are specified in a collection of features applicable to that instance (called the bill of features in ISO/IEC 26580).
- All engineering artefacts that support the creation, design, implementation, deployment, and operation of products are identified and maintained as a single copy of all content used in any system in the product line – i.e. no duplication (called shared asset supersets in ISO/IEC 26580). Content used in all products is common content, which is managed collectively for the product line. Content that varies in one or more system instances is encapsulated with its variations (called the variation points in ISO/IEC 26580), which can be included, omitted, generated, or transformed for a given system instance, based on selected features.
- A system instance is comprised of the variation points, automatically derived according to the selected features for the instance, plus all common content.

5.3 Organizational concepts

5.3.1 Organizations

When an organization, as a whole or a part, enters into an agreement, it is sometimes called a “party” to the agreement. Parties can be from the same organization or from separate organizations. An organization can be as small as a single individual, if the individual is assigned responsibilities and authorities.

In informal terms, the organization that is responsible for executing a process is sometimes referred to by the name of that process. For example, the organization executing the acquisition process is sometimes called the “acquirer”. Other examples include supplier, implementer, maintainer, and operator.

A few other terms are applied to organizations in this document: “user” can be the organization that benefits from the utilization of the product or service; “customer” refers to the user and acquirer collectively; and “stakeholder” refers to an individual or organization with an interest in the system.

The processes and organizations are only related functionally. This document does not dictate or imply a structure for an organization, nor does it specify that particular processes are to be executed by particular parts of the organization. It is the responsibility of the organization that implements this document to define a suitable structure for the organization and assign appropriate roles and responsibilities for the execution of processes.

The processes in this document form a comprehensive set to serve various organizations. An organization, small or large, depending on its purpose or its acquisition strategy, can select an appropriate set of the processes (and associated activities and tasks) to fulfil that purpose. An organization may perform one process or more than one process.

This document is intended to be applied by an organization internally or by two or more organizations. When applied internally, the two agreeing parties typically act under the terms of an agreement that may vary in formality under different circumstances. When applied externally, the two agreeing parties typically act under the terms of a contract. This document uses the term “agreement” to apply to either situation.

For the purpose of this document, any project is assumed to be conducted within the context of an organization. This is important because a project is dependent upon various outcomes produced by the processes of the organization, e.g. employees to staff the project and facilities to house the project. For this purpose, this document provides a set of “organizational project-enabling” processes. These processes are not assumed to be adequate to operate an organization; instead, the processes, considered as a collection, are intended to state the minimum set of dependencies that the project places upon the organization.

5.3.2 Organization and project-level adoption

Modern organizations strive to develop a robust set of life cycle processes that are applied repeatedly to the projects of the organization. Therefore, this document is intended to be useful for adoption at either the organization level or at the project level. An organization would adopt this document and supplement it with appropriate procedures, practices, tools, and policies. A project of the organization would typically conform to the organization's processes rather than conform directly to this document.

In some cases, projects may be executed by an organization that does not have an appropriate set of processes adopted at the organizational level. Such a project may directly adopt the provisions of this document.

5.3.3 Organization and collaborative activities

Due to the increasing complexities of system solutions, it is often useful to employ collaborative and concurrent engineering approaches across the system life cycle. The following are some considerations:

- System life cycle activities are performed in a collaborative manner by involving stakeholders and subject matter experts concurrently, as practical.
- A collaborative framework can be defined within an organization or between organizations to facilitate the involvement of the range of stakeholders and experts. A collaborative framework includes shared methods, tools, and other resources, and establishes an environment for better communication and shared vision and values.
- Collaborative engineering is an essential element of iterative or incremental development approaches to help ensure timely feedback and communication across the stakeholders.

5.4 System of systems concepts

5.4.1 Differences between systems and SoS

An SoS is a set of systems that interact to provide a unique capability that none of the constituent systems can accomplish on its own. In the context of SoS, the relevant pieces of the SoI are, by definition, systems themselves. An SoS consists of some number of constituent systems, plus any inter-system infrastructure, facilities, and processes necessary to enable the constituent systems to integrate or interoperate.

Within an SoS, each constituent system is an independent system that forms part of an SoS. Constituent systems can be part of one or more SoS. Each constituent system is a useful system by itself, having its own development, management, utilization, goals, and resources, but interacts within the SoS to provide the unique capability of the SoS. These additional attributes are what distinguish SoS from a collection of systems.

A system may interact as part of one or more SoS in support of multiple capabilities. In this document, when the interaction of an SoI with an SoS is discussed, this may include one or more SoS in support of one or more capabilities.

The differences between systems and SoS are not in the structure or arrangement of the individual elements, but rather in the behavioural and managerial characteristics of those elements.

5.4.2 Managerial and operational independence

Systems operate within a context of managerial control subject to governance. Organizations govern a portfolio of programmes through goals and objectives, subject to laws, regulations, and external agreements such as contracts. Programmes manage some number of projects to achieve those goals and objectives. Relationships between constituent systems affect the SoS. Systems that do not have any interactions with the constituent systems of a subject SoS are not part of that SoS.

An essential characteristic of the SoS is that constituent systems within the SoS are operationally independent. That is, the constituent systems can (and do) operate independently to fulfil some number of purposes on their own, separate from the SoS. While constituent systems operate independently from each other for their own purposes, they also operate interdependently with each other and other elements to produce the SoS outputs. Constituent systems are never totally independent, yet they are also never totally subservient to the SoS.

Another essential characteristic is that constituent systems within the SoS are both managerially independent and interdependent. Managerial independence suggests that the constituent systems can be managed by organizations that retain some degree of independence even though they are interdependent while participating in SoS. The implication is that these organizations can have goals and objectives for the constituent systems that differ from those of the SoS and the other constituent systems. If so, there is likely some degree of independence and interdependence of governance, as well as some degree of independence and interdependence of management.

Regardless of the means of managing the organizations, alignment (or lack thereof) in the goals and objectives affects the SoS. While some constituent systems are directed or influenced to belong to SoS, some constituent systems can be unaware of the SoS. Some constituent systems choose to belong on a cost/benefits basis, also to cause greater fulfilment of their own purposes, and because of their belief in the overarching SoS purpose.

5.4.3 Taxonomy of SoS

Using essential characteristics to partition the various types of SoS provides an abbreviated nomenclature for thinking about SoS. ISO/IEC/IEEE 21841 defines a normalised taxonomy for systems of systems (SoS) to facilitate communications among stakeholders. It also briefly explains what a taxonomy is and how it applies to the SoS to aid in understanding and communication. There are many characteristics such as scale and scope, around which taxonomies can be derived.

The SoS taxonomy in ISO/IEC/IEEE 21841 organizes the relevant aspects or essential characteristics of SoS, providing specific viewpoints that align with stakeholder concerns. This organization facilitates communications between the various stakeholders that are involved with activities like governance, engineering, operation, and management of these SoS, and provides a reference for other related standards.

ISO/IEC/IEEE 21841 can be useful when describing an SoS or comparing SoS.

5.4.4 SoS considerations in life cycle stages of a system

ISO/IEC/IEEE 21839 provides a set of SoS considerations to be addressed at key points in the life cycle of the SoI. The considerations and key points align with those which are introduced in this document. Selected subsets of these considerations can be applied throughout the life of systems through the involvement of stakeholders. The ultimate goal is to achieve stakeholder satisfaction, so that when delivered, the SoI operates effectively in the operational environment which is typically characterized as one or more systems of systems.

A constituent system can be an entity in more than one SoS. An SoS is often comprised of existing constituent systems along with new constituent systems which are developed and integrated into the SoS. The focus of ISO/IEC/IEEE 21839 is a constituent system as the SoI. The considerations provided in ISO/IEC/IEEE 21839 are with respect to what is necessary to account for the life cycle of the constituent system to enable it to interact in the anticipated SoS configurations.

ISO/IEC/IEEE 21839 can be useful as an augmentation to this document when the SoI is a constituent system within an SoS, which is the typical situation for current systems.

5.4.5 Application of this document to SoS

ISO/IEC/IEEE 21840 provides guidance for the utilization of this document in the context of SoS. While this document applies to systems in general (including constituent systems), ISO/IEC/IEEE 21840 provides guidance on the application of these processes to the special case of SoS. However, ISO/IEC/IEEE 21840 is not a self-contained SoS replacement for this document. ISO/IEC/IEEE 21840 is intended to be used in conjunction with this document, ISO/IEC/IEEE 21839, and ISO/IEC/IEEE 21841 and is not intended to be used as standalone guidance.

When the SoI is part of an SoS, the application of the processes in this document is dependent on the type of SoS life cycle and impact of the SoS interactions on the SoI. In some cases, there can be “waves” of SoS revision. In other cases, SoS changes occur continually, with many processes operating on a continuous basis to implement evolutionary change.

5.5 Life cycle concepts

5.5.1 System life cycle model

Every system has a life cycle. A life cycle can be described using an abstract functional model that represents the conceptualization of a need for the system, its realization, utilization, evolution, and disposal.

A system progresses through its life cycle as the result of actions, performed and managed by people in organizations, using processes for execution of these actions. The detail in the life cycle model is expressed in terms of these processes, their outcomes, relationships, and sequence. This document does not prescribe any particular life cycle model. Instead, it defines a set of processes, termed life cycle processes, that can be used in the definition of the system's life cycle. The processes described in this document support sequential as well as iterative and incremental development models. Also, this document does not prescribe any particular sequence of processes within the life cycle model. The sequence of the processes is determined by project objectives and by selection of the system life cycle model.

5.5.2 System life cycle stages

Life cycles vary according to the nature, purpose, use, and prevailing circumstances of the system. Each stage has a distinct purpose and contribution to the whole life cycle and is considered when planning and executing the system life cycle. The life cycle model comprises one or more stages, as needed. It is assembled as a sequence of stages that can be iterative, concurrent, or overlapping, as appropriate for the SoI's scope, magnitude, complexity, changing needs, and opportunities.

The stages represent the major life cycle periods associated with a system and they relate to the state of the system description or the system itself. The stages describe the major progress and achievement milestones of the system through its life cycle. They give rise to the primary decision gates of the life cycle. These decision gates apply specific decision criteria and are used by organizations to understand and manage the inherent uncertainties and risks associated with business case, costs, schedule, performance, or functionality of a system. The stages thus provide a framework within which organization management has high-level visibility and control of technical management and technical processes. ISO/IEC/IEEE 24748-1 describes the application of these processes in any stage, provides more details on decision gates, and defines typical system life cycle stages, including

- concept;
- development;
- production;

- utilization;
- support;
- retirement.

Note that there are significant differences between life cycle stages (e.g. utilization, support, retirement) and processes (e.g. operation, maintenance, disposal).

Organizations employ stages differently to satisfy contrasting strategies. The life cycle stages often occur concurrently, especially in continuous, incremental, or evolutionary approaches. Using stages concurrently and in different orders can lead to life cycle forms with distinctly different characteristics.

Further elaboration of these concepts can be found in ISO/IEC/IEEE 24748-1 and ISO/IEC/IEEE 24748-2, which provide guidelines on the application of life cycle management.

NOTE The set of events, occurrences, and evolution across the life cycle is sometimes referred to as the system life history. Per ISO 15704, the life history is the actual, recorded and configuration managed sequence of steps the system has gone through during its lifetime.

5.6 Process concepts

5.6.1 Criteria for processes

The determination of the life cycle processes in this document is based upon three basic principles:

- Each life cycle process has strong relationships among its outcomes, activities, and tasks.
- The dependencies among the processes are reduced to the greatest feasible extent.
- A process is capable of execution by a single organization in the life cycle.

5.6.2 Description of processes

Each process of this document is described in terms of the following attributes:

- The title conveys the scope of the process as a whole.
- The purpose describes the goals of performing the process.
- The outcomes express the observable results expected from the successful performance of the process.
- The activities are sets of cohesive tasks of a process.
- The tasks are requirements intended to support the achievement of the outcomes.

Additional detail regarding this form of process description can be found in ISO/IEC/IEEE 24774. Outputs are an optional attribute for a process description. They are artefacts or information items. [Annex B](#) includes examples of outputs from the processes.

5.6.3 General characteristics of processes

In addition to the basic attributes described in [5.6.2](#), processes may be characterized by other attributes common to all processes. The ISO/IEC 33000 family of standards identifies common process attributes that characterize six levels of achievement within a measurement framework for process capability.

5.7 Processes in this document

5.7.1 General

This document groups the activities that can be performed during the life cycle of a system into four process groups:

- a) agreement processes;
- b) organizational project-enabling processes;
- c) technical management processes;
- d) technical processes.

The groups and the processes included in each group are depicted in [Figure 4](#). Each of the life cycle processes is described in terms of its purpose and desired outcomes with a set of related activities and tasks that can be performed to achieve those outcomes.

The processes described in this document are not intended to preclude or discourage the use of additional processes that organizations find useful. The order of the subclauses in which the processes are defined in this document does not determine the order in which the processes are performed during the system life cycle or any of its stages (i.e. there is no prescriptive order or sequence). A description of each process group is provided in [5.7.2](#) to [5.7.5](#).

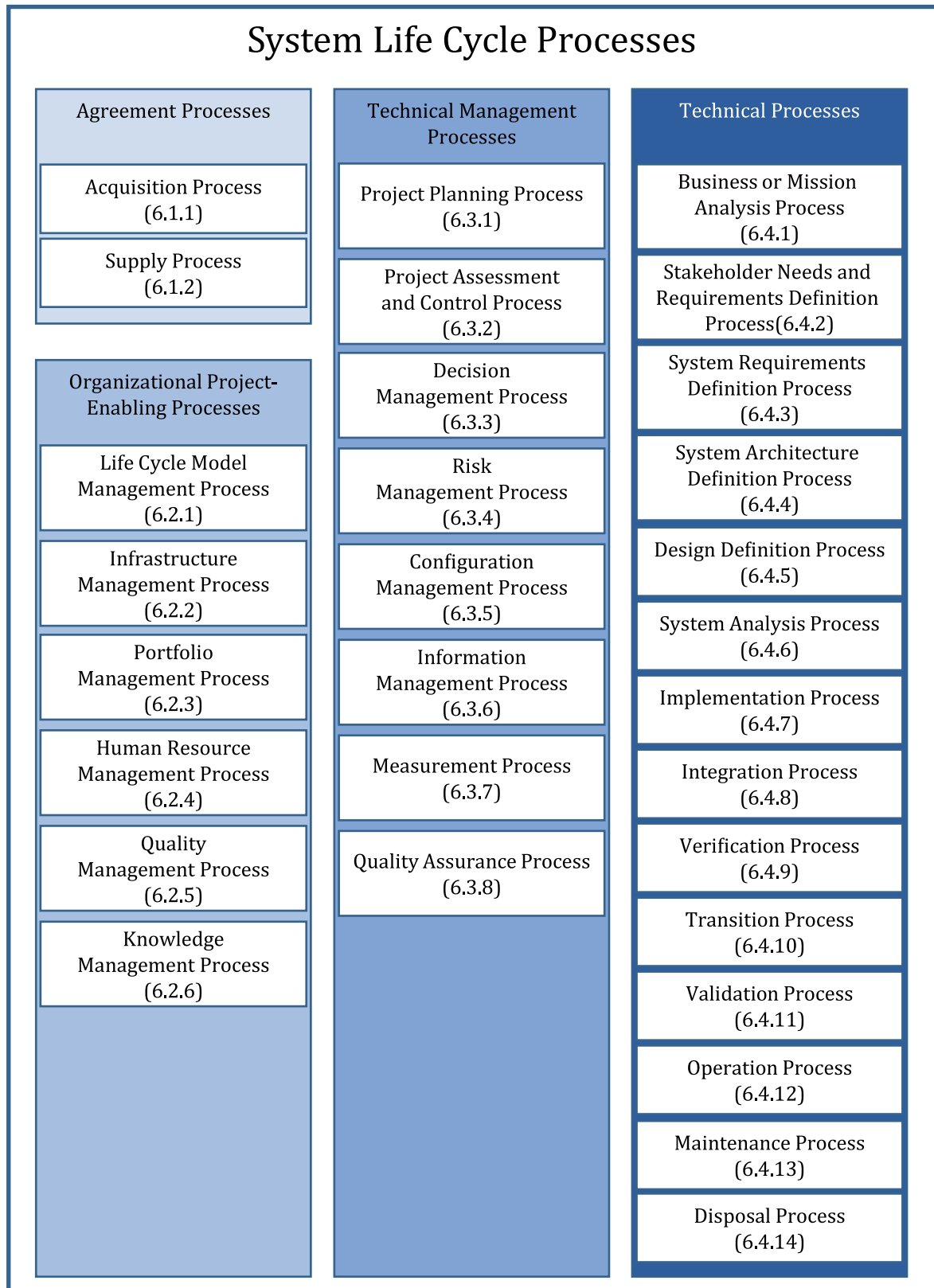


Figure 4 — System life cycle processes

5.7.2 Agreement processes

Organizations are producers and users of systems. One organization (acting as an acquirer) can task another (acting as a supplier) for products or services. This is achieved using agreements. Agreements allow both acquirers and suppliers to realise value and support strategies for their organizations.

Generally, organizations act simultaneously or successively as both acquirers and suppliers of systems. The agreement processes can be used with less formality when the acquirer and the supplier are in the same organization. Similarly, they can be used within the organization to agree on the respective responsibilities of organization, project, and technical functions.

The agreement processes consist of the following (also see [Figure 4](#)):

- a) acquisition process – used by organizations for acquiring products or services;
- b) supply process – used by organizations for supplying products or services.

These processes define the activities necessary to establish an agreement between two organizations. If the acquisition process is invoked, it provides the means for interacting with a supplier. This may include products that are supplied for use as an operational system, services in support of operational activities, or elements of a system being provided by a supplier. If the supply process is invoked, it provides the means for an agreement for a product or service that is provided to the acquirer.

NOTE 1 Security is an increasing concern in systems engineering. See the ISO/IEC 27036 series for requirements and guidance for suppliers and acquirers on how to secure information in supplier relationships. Specific aspects of information security supplier relationships are addressed in ISO/IEC 27036-3 and ISO/IEC 27036-4.

When an SoI participates as part of an SoS, it is often necessary to consider the resource and capability dependencies in the performance of the agreement processes. As needed, the agreements would include clauses for the SoS interactions and dependencies or additional agreements generated. The processes apply for the agreements between the stakeholders of the SoI and interoperating systems. If there is an external entity with some type of responsibility that spans an SoS in which the SoI is a constituent system, then management and support arrangements can be required with that external entity. The agreements establish the responsibilities and modes of support and control across the life cycle stages among the stakeholder organizations from the context of the SoI participation in the SoS. This is of greater importance if the organization holds primary objectives for their constituent system that may not be directly aligned with those of the SoS.

NOTE 2 More information about process application for the SoS is provided in ISO/IEC/IEEE 21839 and ISO/IEC/IEEE 21840.

5.7.3 Organizational project-enabling processes

The organizational project-enabling processes are concerned with providing the resources needed to enable the project to meet the needs and expectations of the organization's stakeholders. The organizational project-enabling processes are typically concerned at a strategic level with the management and improvement of the organization's undertaking, with the provision and deployment of resources and assets, and with its management of risks in competitive or uncertain situations.

The organizational project-enabling processes establish the environment in which projects are conducted. The organization establishes the processes and life cycle models to be used by projects; establishes, redirects, or cancels projects; provides resources required, including human and financial; and sets and monitors the quality measures for systems and other deliverables that are developed by projects to satisfy internal and external customers.

The organizational project-enabling processes do not necessarily imply commercial or profit-making motives. Organizational project-enabling processes are equally relevant to non-profit organizations, since they are also accountable to stakeholders, are responsible for resources and encounter risk in their undertakings. This document can be applied to non-profit organizations as well as to profit-making organizations. In addition, organizational project-enabling processes are not intended to be a comprehensive set of organizational processes that enable strategic management of the organization.

The organizational project-enabling processes consist of the following (also see [Figure 4](#)):

- a) life cycle model management process;
- b) infrastructure management process;
- c) portfolio management process;
- d) human resource management process;
- e) quality management process;
- f) knowledge management process.

In a typical SoI, organizational project-enabling processes establish the environment and provide the necessary resources for the conduct of projects to address system solutions. The organization establishes the processes and life cycle models to be used by projects; establishes, redirects, or cancels projects; provides resources required, including human, material and financial; and sets and monitors the quality measures for systems and other deliverables that are developed by projects for internal and external customers. These processes also provide the environment and resources for the SoI to be able to support capabilities provided by an SoS in which the SoI participates. The organizations responsible for the constituent systems implement these processes for their SoI independent of the SoS. These processes can be influenced by regulations, interface standards, or agreements for those parts of the SoI that contribute to the overall SoS capabilities.

NOTE More information about process application for the SoS is provided in ISO/IEC/IEEE 21839 and ISO/IEC/IEEE 21840.

5.7.4 Technical management processes

The technical management processes are concerned with managing the resources and assets allocated by organization management and with applying them to fulfil the agreements into which the organization or organizations enter. The technical management processes relate to the technical effort of projects, in particular to planning in terms of cost, timescales, and achievements; to the checking of actions to help ensure that they comply with plans and performance criteria; and to the identification and selection of corrective actions that recover shortfalls in progress and achievement. These processes are used to establish and perform technical plans for the project, manage information across the technical team, assess technical progress against the plans for the system products or services, control technical tasks through to completion, and to aid in the decision-making process. Individual technical management processes may be invoked at any time in the life cycle and at any level in a hierarchy of projects, as required by plans or unforeseen events. The technical management processes are applied with a level of rigor and formality that depends on the agreements as well as risk and complexity of the project.

NOTE 1 Technical management is 'the application of technical and administrative resources to plan, organize, and control engineering functions' (ISO/IEC/IEEE 24765).

Typically, several projects co-exist in any one organization. The technical management processes can be employed at a corporate level to meet internal needs.

NOTE 2 Technical management processes are applied during the performance of each technical process.

The technical management processes can be applied to manage the technical activities through the life cycles of systems, including products or services.

NOTE 3 This set of technical management processes are performed so that system-specific technical processes can be conducted effectively. They do not comprise a management system or a comprehensive set of processes for project management, as that is not within the scope of this document.

The technical management processes consist of the following (also see [Figure 4](#)):

- a) project planning process;

- b) project assessment and control process;
- c) decision management process;
- d) risk management process;
- e) configuration management process;
- f) information management process;
- g) measurement process;
- h) quality assurance process.

Project planning and project assessment and control are key to all management practices. These processes establish the general approach for managing a project or a process. The other processes in this group provide a specific focused set of tasks for performing to a specialised management objective. They are all evident in the management of any undertaking, ranging from a complete organization down to a single life cycle process and its tasks. In this document, the project is chosen as the context for describing processes. The same processes can also be applied in the performance of services.

Technical management processes are concerned with managing the resources and assets allocated by organization management and with applying them to fulfil the agreements into which the organization or organizations enter. The technical management processes need to include the considerations of the expected SoS interactions. The considerations span the planning, assessment, and management of resources, risks, and other factors associated with dependencies from other interacting systems. For example, the planning, assessment, and control activities need to include SoS-related cost and schedule considerations for the SoI. This includes monitoring the progress in the areas of cross-system dependencies. The agreement processes are executed to negotiate the required resources across stakeholders.

NOTE 4 More information about process application for the SoS is provided in ISO/IEC/IEEE 21839 and ISO/IEC/IEEE 21840.

5.7.5 Technical processes

The technical processes are concerned with technical actions throughout the life cycle. Technical processes transform the needs of stakeholders into products or services. By applying that product or operating that service, technical processes provide sustainable performance, when and where needed, to meet the stakeholder requirements and achieve customer satisfaction. The technical processes are applied to create and use a system, whether it is in the form of a model or is a finished product.

The technical processes are used to define the requirements for a system, to transform the requirements into an effective product, to permit consistent reproduction of the product where necessary, to use the product, to provide the required services, to sustain the provision of those services and to dispose of the product when it is retired from service.

The technical processes define the activities that enable organization and project functions to optimise the benefits and reduce the risks that arise from technical decisions and actions. These activities enable products and services to possess the timeliness and availability, the cost effectiveness, and the functionality, reliability, maintainability, producibility, usability and other quality characteristics required by acquiring and supplying organizations. They also enable products and services to conform to the expectations, ethical perspectives, or legislated requirements of society.

The technical processes consist of the following (also see [Figure 4](#)):

- a) business or mission analysis process;
- b) stakeholder needs and requirements definition process;
- c) system requirements definition process;

- d) system architecture definition process;
- e) design definition process;
- f) system analysis process;
- g) implementation process;
- h) integration process;
- i) verification process;
- j) transition process;
- k) validation process;
- l) operation process;
- m) maintenance process;
- n) disposal process.

NOTE 1 For software and hardware system elements, these processes are applied at recursively lower levels for system definition and recursively higher levels for system realization.

NOTE 2 These processes are often performed concurrently, iterating between one another to establish a solution that is balanced with respect to requirements, critical performance measures, critical quality characteristics, and SoS considerations. At any level of abstraction, system requirements and models are made consistent via iterations of applicable technical processes. When requirements and models are not directly capable of being implemented, the same processes are repeated recursively throughout the system structure.

NOTE 3 Interface management is a set of activities that cut across the systems engineering processes. These are cross-cutting activities of the technical and technical management processes that apply and track as a specific view of the processes and system. See ISO/IEC/IEEE 24748-1 for an example interface management process view and INCOSE-TP-2003-002-5, Part III, section 3.2.4 for more information.

Technical processes are concerned with technical actions throughout the life cycle of the SoI. As the technical processes are performed for the provision and support of technical solutions across the life cycle stages, SoS considerations for the SoI include the technical impact on interacting systems and their stakeholders and infrastructure. ISO/IEC/IEEE 21839 states that “this includes both systems/services on which the SoI depends and systems/services that depend on the SoI”. This can impose new requirements or constraints on the SoI by the SoS configurations in which the SoI participated in support of required or desired business or mission capabilities. The SoS technical considerations apply to all the technical processes across the life cycle stages and play an especially important role in the business or mission analysis, stakeholder needs and requirements definition, system requirements definition, system architecture definition, and design definition processes.

NOTE 4 ISO/IEC/IEEE 21839 and ISO/IEC/IEEE 21840 provide more information on process application for the SoS.

5.8 Process application

5.8.1 Overview

The life cycle processes defined in this document can be used by any organization when acquiring, using, creating, modifying or supplying a system. They can be applied at any point in a system’s structure and at any stage in the life cycle.

The functions these processes perform are defined in terms of specific purposes, outcomes, and the set of activities and tasks that constitute the process.

Each life cycle process in [Figure 4](#) can be invoked, as required, at any time throughout the life cycle. The application of these processes is influenced by many factors throughout the life cycle of the system, which may require the processes to be applied in an iterative, recursive, or concurrent manner.

[Figure 5](#) illustrates the interrelationships among processes defined in this document. The technical management processes are continually applied to manage and control all of the processes and life cycle stages. The system analysis process provides data and information that are essential for each iteration of the systems life cycle processes, which are performed to support any of the technical management processes ([6.3](#)) and the technical processes ([6.4](#)).

NOTE The arrows in [Figure 5](#) are intended to show general relationships that include iterative, recursive, and concurrent application. All possible relationships are not included. The actual flows or interactions between the processes for a project are determined by the project tailoring and needs. Also, the arrows are not intended to indicate any specific temporal relationships, sequences, or scheduling. The arrows between process groups or aggregations of processes within process groups are intended to indicate that the project can apply the processes in any order, can iterate between processes, and can implement them concurrently.

The changing nature of the influences on the system (e.g. operational environment changes, new opportunities for system element implementation, modified structure, and responsibilities in organizations) requires continual review of the selection and timing of process use. Process use in the life cycle can be dynamic, responding to the many external influences on the system or internal influences from a more continuous development approach. The life cycle approach also allows for incorporating the changes in the next stage. The life cycle stages assist the planning, execution, and management of life cycle processes in the face of this complexity in life cycles by providing comprehensible and recognizable high-level purpose and structure. The set of processes within a life cycle stage are applied with the common goal of satisfying the exit criteria for that stage or the entry criteria of the formal progress reviews within that stage. This applies regardless of the type of life cycle model or development approach.

Where justified by quality risks, detailed descriptions of process instances in the context of the specific product or service may also be created. Instantiation of processes involves identifying specific success criteria for a process instance, derived from the requirements, and identifying the specific activities and tasks needed to achieve the success criteria, derived from the activities and tasks identified in this document. Creating detailed descriptions of process instances enables better management of quality risks by establishing the link between the process and the specific requirements.

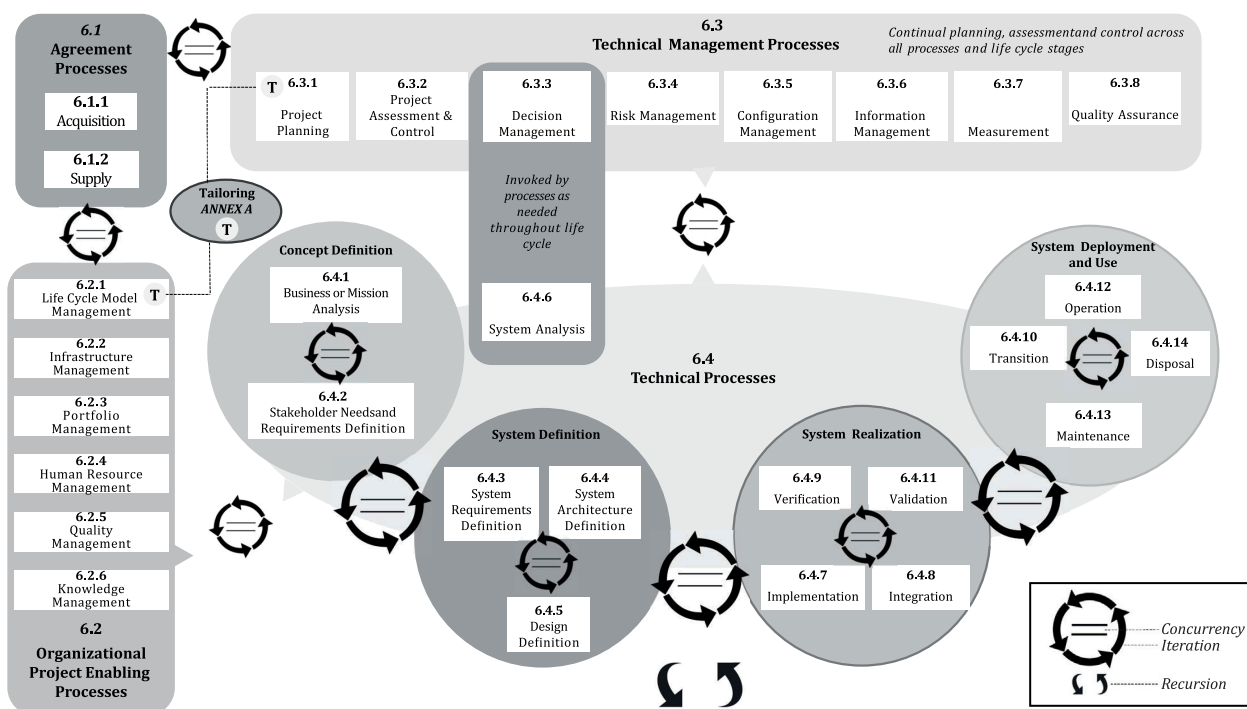


Figure 5 — Interrelationships between processes

The processes in this document are often applied using an MBSE approach, which uses a set of models to implement the processes and achieve the expected outcomes. See [Annex D](#) for information regarding MBSE.

5.8.2 Process iteration, recursion, and concurrency

When the application of the same process or set of processes is repeated on the same level of the system structure, the application is referred to as iterative. The iterative use of processes is important for the progressive refinement of process outputs, for example, the interaction between successive verification actions and integration actions can incrementally build confidence in the conformance of the product or service. Iteration is not only appropriate but also expected. New information is created by the application of a process or set of processes. Typically, this information takes the form of questions with respect to requirements, analysed risks, or opportunities. Iterative application of a process or set of processes should continue until such questions are resolved.

Iteration between business or mission analysis, stakeholder needs and requirements definition, system requirements definition, system architecture definition, and design definition processes often occurs to help achieve a common understanding of the problem to be solved and the identification of a satisfactory solution. These are heavily supported by the system analysis and decision management processes (see [Figure 5](#)). Information produced by the processes should be shared and used by all other system life cycle processes.

The recursive use of processes, i.e. the repeated application of the same process or set of processes applied to successive levels of system elements in a system's structure, is a key aspect of the application of this document. From a view of relations between a system and its system elements, the outputs of processes applied for a system or system element, whether information, artefacts, or services, are inputs to other processes or other system elements for additional analysis or more generalised synthesis of its system elements, to arrive at a more detailed or mature set of outcomes. Such an approach adds value to systems at successive levels in the system structure.

The discussion in this subclause on iterative and recursive use of the system life cycle processes is not meant to imply any specific hierarchical, vertical, or horizontal structure for the SoI, enabling system, organization, or project.

Concurrent use of processes can exist within a project (e.g. when design actions and preparatory actions for building a system are performed at the same time), and between projects (e.g. when system elements are designed at the same time under different project responsibilities). All processes can be used in parallel with other processes. As an example, the operation and maintenance processes need to provide input to the system requirements definition, system architecture definition, design definition, and implementation processes.

5.8.3 Process views

There are cases where a unified focus is needed for activities and tasks that are selected from disparate processes to provide visibility to a significant concept or thread that cuts across the processes employed across the life cycle. For this purpose, the concept of a process view has been formulated. Like a process, the description of a process view includes a statement of purpose and outcomes. Unlike a process, the description of a process view does not include unique activities and tasks. Instead, the description includes guidance explaining how the outcomes can be achieved by employing the activities and tasks of the various life cycle processes. The detailed information about process views is available in ISO/IEC/IEEE 24748-1 and ISO/IEC/IEEE 24774. The following International Standards provide details on some of the technical viewpoints:

- ISO/IEC/IEEE 24748-1
 - specialty engineering;
 - interface management;

- security.
- ISO/IEC/IEEE 15026-4
- system assurance;
- software assurance.

5.9 Concept and system definition

Concepts, needs, and requirements evolve at various levels resulting from the establishment of or changes to the organizational concept of operations, strategy, or environment. The concept of operations addresses the leadership's intended way of operating the organization. This evolution is accomplished through the application of the business or mission analysis, stakeholder needs and requirement definition, system requirements definition, system architecture definition, and design definition processes with the support of other processes as needed. Through the iterative and concurrent use of all these processes, stakeholders are identified and insights are gained into the relationships between the concepts, needs, and requirements of the organization, stakeholders, and systems, as well as the emergent properties and behaviours of the system that arise from the interactions and relations among the system elements and environment.

The business or mission analysis process analyses changes in the organizational concept of operations, environment, and other strategic inputs to identify and define key problems or opportunities that should be addressed to achieve the organization's mission(s), vision, goals, or objectives. The business or mission analysis process also identifies, characterizes, and prioritises alternative solution classes (or general approaches) that are candidates to address the problem or opportunity. Using the stakeholder needs and requirements definition process, stakeholders define their concepts, needs, and requirements in the context of the defined problem or opportunity, the associated capabilities required, and the preferred solution class(es). This includes defining the operational context of the solution. The operational concept addresses what the system will do and why. Using the system requirements definition process, the engineering team tasked with providing a solution transform the stakeholder requirements into system requirements. In the application of the three processes discussed in this document, a range of analysis techniques and trade-offs are applied iteratively and recursively to transform concepts into needs and needs into requirements (e.g. mission analysis, business analysis, operational analysis, requirements analysis).

NOTE 1 ISO/IEC/IEEE 29148 provides further details on the development of concepts, needs, and requirements. It includes lower-level elaboration for the processes discussed in this document, as well as annotated outlines for documenting operational concepts, stakeholder needs and requirements, and system requirements.

The system architecture definition process focuses on defining an architecture that addresses the stakeholder concerns and is applied iteratively and concurrently with the business or mission analysis, stakeholder needs and requirements definition, and system requirements definition processes to determine the best solution to address stakeholder concerns. The design definition process, on the other hand, is driven by requirements that have been vetted through evaluation with the architecture and more detailed analyses of feasibility. Architecture focuses on suitability, viability, and desirability, whereas design focuses on compatibility with technologies and other design elements and on feasibility of construction and integration. An effective architecture is as design-agnostic as possible to allow for maximum flexibility in the design trade space.

The design definition process provides feedback to the system architecture definition process to consolidate or confirm the allocation, partitioning and alignment of architectural entities (e.g. strategic goals, capabilities and effects, operational activities, resource functions) to system elements that comprise the system. Note that an architecture entity (an entity being architected or one subject to the

system architecture definition process) and a system element (a discrete part of a system that fulfils specified requirements) represent two different notions.

NOTE 2 Practices, conventions, principles, and concepts for system architecture definition are specified by the ISO/IEC/IEEE 420x0 family of standards. ISO/IEC/IEEE 42020 provides architecture processes for the governance, management, conceptualization, evaluation, and elaboration of architectures; ISO/IEC/IEEE 42030 provides an architecture evaluation framework for performing architectural analysis, value assessment, and evaluation synthesis; and ISO/IEC/IEEE 42010 provides key principles and concepts for describing an architecture.

The enterprise architecture(s) or relevant reference architectures, when available, can provide useful insights for the system architecture through the life cycle of the SoI. Additionally, when the organization or enterprise is treated as the SoI, the enterprise architecture becomes a relevant part of the system definition, since it is then the top-level system architecture.

The processes discussed in this clause interact with other technical and technical management processes to provide necessary inputs and information. For example, the system analysis process provides analysis results to support trade-offs that are managed by the decision management process. Additionally, the establishment of the concepts, requirements, architecture, and design are informed by the other technical processes, such as the operations and maintenance processes.

5.10 Assurance and quality characteristics

Assurance is defined as grounds for justified confidence that a claim has been or will be achieved. This confidence is achieved by applying applicable system life cycle activities, which include a planned, systematic approach with acceptable measures of system assurance and risk management of exploitable vulnerabilities. Stakeholders need assurance prior to depending on a system, especially a system involving complexity, novelty, or technology with a history of problems. The greater the degree of dependence, the greater the need for strong assurance. System assurance claims frequently concern the functions or capabilities of the system.

Assurance, as for a variety of attributes such as safety, security, and dependability, is often required. Stakeholder concerns include achieving justified confidence that the system, while achieving its intent, does not also provide unintended behaviour or produce unintended outcomes. A claims-oriented approach to assurance serves to address the concerns that are not typically captured within the requirements that focus on intended behaviour. An assurance case can identify gaps in requirements coverage and inform the development of derived requirements to address those gaps.

Assurance is often provided through activities to construct an assurance case. An assurance case is an auditable artefact that provides a convincing and valid argument for a claim on the basis of tangible evidence under a given context. Subtle and complex arguments are necessary to organize a wide variety and huge amount of evidence and link them to the claim. Pieces of evidence can be pass/fail results of test, quantitative measurements, or qualitative evaluations. These pieces require careful review on their validity, certainty, fairness, etc., when they are integrated into a cohesive argument. Often there is not a simple direct connection between the evidence provided and the overall claim, so there has to be a structure to describe the subclaims and the reasoning that links this overall claim and the evidence.

Assurance cases focusing on specific characteristics include safety case, composed assurance package (for security), and dependability case.

Assurance activities should be integrated into life cycle processes throughout the system life cycle. Construction of assurance cases require detailed knowledge about both the SoI and the characteristics under consideration. Integration of the specific analyses in the SoI development and involving experts in the SoI development are key success factors to achieve a compliant and balanced system solution.

NOTE The ISO/IEC/IEEE 15026 series provides more information on systems and software assurance and assurance cases.

Considerations for requirements on specific characteristics are provided by other documents including the IEC 61508 series (safety); ISO/IEC 27000 and ISO/IEC 15408-3 (information security); IEC 60300-1

and IEC 62741 (dependability); and ISO/IEC 25000 (systems and software quality requirements and evaluation).

5.11 Process reference model

[Annex C](#) defines a process reference model for processes contained in [Clause 6](#). The process reference model is applicable to an organization that is assessing its processes to determine the capability of these processes. The purpose and outcomes are a statement of the goals of the performance of each process. This statement of goals permits assessment of the effectiveness of the processes in ways other than simple conformity assessment.

6 System life cycle processes

6.1 Agreement processes

6.1.1 Acquisition process

6.1.1.1 Purpose

The purpose of the acquisition process is to obtain a product or service in accordance with the acquirer's requirements.

NOTE As part of this process, the agreement is modified when a change request is agreed to by both the acquirer and supplier.

6.1.1.2 Outcomes

As a result of the successful performance of the acquisition process:

- a) a request for supply is prepared;
- b) one or more suppliers are selected;
- c) an agreement is established between the acquirer and supplier;
- d) a product or service complying with the agreement is accepted;
- e) acquirer obligations defined in the agreement are satisfied;
- f) responsibility for the acquired product or service is transferred, as directed by the agreement.

6.1.1.3 Activities and tasks

The acquirer shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the acquisition process.

NOTE 1 The activities and resulting agreement from this process often apply to suppliers in the supply chain, including subcontracted suppliers.

- a) Prepare for the acquisition. This activity consists of the following tasks.
 - 1) Define a strategy for how the acquisition will be conducted.

NOTE 2 This strategy describes or references the life cycle model, risks and issues mitigation, a schedule of milestones and decision gates, and selection criteria if the supplier is external to the acquiring organization. It also includes key drivers and characteristics of the acquisition, such as responsibilities and liabilities; specific models, methods, or processes; level of criticality; formality; and priority of relevant trade-off factors.

- 2) Prepare a request for the supply of a product or service that includes the requirements.

NOTE 3 If a supplier is external to the organization, then the request typically includes the practices with which a supplier is expected to comply and the criteria for selecting a supplier.

NOTE 4 A definition of requirements is provided to one or more suppliers. The requirements are the stakeholder or the system requirements, depending on the type of acquisition approach, through the associated requirements definition process.

- b) Advertise the acquisition and select the supplier. This activity consists of the following tasks.

- 1) Communicate the request for the supply of a product or service to potential suppliers.

- 2) Select one or more suppliers.

NOTE 5 To obtain competitive solicitations, proposals to supply are evaluated and compared against the selection criteria and ranked. The justification for rating each proposal is declared and suppliers are informed why they were or were not selected.

- c) Establish and maintain an agreement. This activity consists of the following tasks.

NOTE 6 Project cost, schedule, and performance are monitored through the project assessment and control process. Any identified issues that require agreement modifications are referred to this activity. Any proposals for changes to system elements or information are controlled through the change management activity of the configuration management process.

NOTE 7 For an SoS, if a multi-lateral agreement exists, responsibilities and modes of support and control are established across the life cycle stages among the stakeholder organizations participating in the SoS. Agreements are flexible to adapt to changing requirements of an SoS for which the SoI is a constituent system.

- 1) Develop and approve an agreement with the supplier that includes acceptance criteria.

NOTE 8 This agreement ranges in formality from a written contract to a verbal agreement. Appropriate to the level of formality, the agreement establishes requirements, development and delivery milestones, verification, validation and acceptance conditions, process requirements (e.g. configuration management, risk management, measurement), exception-handling procedures, agreement change management procedures, payment schedules, accountability of each party in case of non-fulfilment, and handling of data rights and intellectual property so that both parties of the agreement understand the basis for executing the agreement. For a written contract, this occurs when the contract is signed.

NOTE 9 The agreement identifies any requirements to be imposed on participating subcontractors.

- 2) Identify necessary changes to the agreement.

NOTE 10 In requesting a change to the agreement, the acquirer or the supplier details its specifications, rationale, and background.

- 3) Evaluate impact of changes on the agreement.

NOTE 11 Any change is investigated for impacts to project plans, schedule, cost, technical capability, and quality. A change can be handled within the existing agreement, can require a modification to the agreement, or can require a new agreement.

- 4) Update the agreement with the supplier, as necessary.

NOTE 12 The result of the agreement modification is incorporated into the project plans and communicated to all affected parties.