



DPC: 25/30484184 DC

BSI Group Headquarters

389 Chiswick High Road London W4 4AL

Tel: +44 (0)20 8996 9000

Fax: +44 (0)20 8996 7400

www.bsigroup.com

Date: 13 November 2025

Origin: International

Latest date for receipt of comments: 06 January 2026

Project No. 2023/02933

Responsible committee: IST/15 Software and systems engineering

Interested committees:

Title: Draft BS ISO/IEC/IEEE 42024 Enterprise, systems and software — Architecture fundamentals.

Please notify admin.start@bsigroup.com if you are aware of any keywords that might assist in clarifying or identifying the standard or of the content of this standard:

- i) has any issues related to 3rd party IPR, patent or copyright
- ii) affects other national standard(s)
- iii) requires additional national guidance or information

WARNING: THIS IS A DRAFT AND MUST NOT BE REGARDED OR USED AS A BRITISH STANDARD. THIS DRAFT IS NOT CURRENT BEYOND 06 January 2026

This draft is issued to allow comments from interested parties; all comments will be given consideration prior to publication. No acknowledgement will normally be sent. **See overleaf for information on the submission of comments.**

No copying is allowed, in any form, without prior written permission from BSI except as permitted under the Copyright, Designs and Patent Act 1988 or for circulation within a nominating organization for briefing purposes. Electronic circulation is limited to dissemination by e-mail within such an organization by committee members.

Further copies of this draft may be purchased from BSI Shop <http://shop.bsigroup.com> or from BSI Customer Services, Tel: +44(0) 20 8996 9001 or email cservices@bsigroup.com. British, International and foreign standards are also available from BSI Customer Services.

Information on the co-operating organizations represented on the committees referenced above may be obtained from <https://standardsdevelopment.bsigroup.com/>.

E-mail: admin.start@bsigroup.com

Introduction

This draft standard is based on national and international discussions. Your comments on this draft are invited and will assist in the preparation of the consequent standard.

For international standards, comments will be reviewed by the relevant UK national committee before sending the consensus UK vote and comments to the international committee, which will then decide appropriate action. If the international standard is approved, it is usual for the text to be published as a British Standard.

For national standards, comments will be reviewed by the relevant UK national committee and the resulting standards published as a British Standard.

UK Vote

Please indicate whether you consider the UK should submit a negative (with supporting technical reasons) or positive vote on this draft. Please indicate if you are aware of any reason why this draft standard should not be published as a British Standard.

Submission of Comments

- **Annotated drafts are not acceptable and will be rejected.**
- All comments should be submitted online at <https://standardsdevelopment.bsigroup.com/>. You will need to register in order to comment.

Template for comments and secretariat observations

Date: xx/xx/20xx	Document: ISO/DIS xxxx
------------------	------------------------

1	2	(3)	4	5	(6)	7
MB	Clause No./ Subclause No./Annex (e.g. 3.1)	Paragraph/ Figure/Table/Note	Type of comment	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
	3.1	Definition 1	ed	Definition is ambiguous and needs clarifying.	Amend to read '...so that the mains connector to which no connection...'	
	6.4	Paragraph 2	te	The use of the UV photometer as an alternative cannot be supported as serious problems have been encountered in its use in the UK.	Delete reference to UV photometer.	



DRAFT International Standard

ISO/IEC/IEEE DIS 42024

Enterprise, systems and software — Architecture fundamentals

ICS: 35.080

ISO/IEC JTC 1/SC 7

Secretariat: **BIS**

Voting begins on:
2025-11-13

Voting terminates on:
2026-02-05

This document is circulated as received from the committee secretariat.

THIS DOCUMENT IS A DRAFT CIRCULATED FOR COMMENTS AND APPROVAL. IT IS THEREFORE SUBJECT TO CHANGE AND MAY NOT BE REFERRED TO AS AN INTERNATIONAL STANDARD UNTIL PUBLISHED AS SUCH.

IN ADDITION TO THEIR EVALUATION AS BEING ACCEPTABLE FOR INDUSTRIAL, TECHNOLOGICAL, COMMERCIAL AND USER PURPOSES, DRAFT INTERNATIONAL STANDARDS MAY ON OCCASION HAVE TO BE CONSIDERED IN THE LIGHT OF THEIR POTENTIAL TO BECOME STANDARDS TO WHICH REFERENCE MAY BE MADE IN NATIONAL REGULATIONS.

RECIPIENTS OF THIS DRAFT ARE INVITED TO SUBMIT, WITH THEIR COMMENTS, NOTIFICATION OF ANY RELEVANT PATENT RIGHTS OF WHICH THEY ARE AWARE AND TO PROVIDE SUPPORTING DOCUMENTATION.

Reference number
ISO/IEC/IEEE DIS 42024:2025(en)

Full document details: 25/30484184 DC

© ISO/IEC 2025
© IEEE 2025

Documents
Document Sharing Hub



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2025

© IEEE 2025

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO or IEEE at the respective address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Institute of Electrical and Electronics Engineers, Inc
3 Park Avenue, New York
NY 10016-5997, USA

Email: stds.ipr@ieee.org
Website: www.ieee.org

Contents

Page

Foreword..... v

Introduction..... vi

1 Scope..... 1

2 Normative references..... 1

3 Terms and definitions..... 1

3.1 Index of terms..... 2

3.2 Architecting concepts..... 3

3.3 Entity related concepts..... 4

3.4 Architecture design concepts..... 6

3.5 Architecture description concepts..... 8

3.6 Entity life cycle concepts..... 9

4 Architecture concepts..... 10

4.1 Overview..... 10

4.2 Architecture and architecting..... 10

4.2.1 Introduction..... 10

4.2.2 Architecting effort..... 11

4.2.3 Architecture features..... 12

4.2.4 Fundamental concepts and essential properties..... 13

4.2.5 Architecting principles and architecture principles..... 14

4.3 Architecture design and solution design..... 15

4.3.1 Architecting effort to produce an architecture design..... 15

4.3.2 Outcomes of an architecting effort..... 17

4.3.3 Primary entity of interest and related entities of interest..... 18

4.3.4 Architecture alternatives investigation..... 18

4.3.5 Environment and context..... 18

4.4 Architecture definition and communication..... 19

4.4.1 Architecture information package contents..... 19

4.4.2 Architecture and solution design information characteristics..... 20

4.4.3 Engineering effort to produce a solution design..... 21

4.4.4 Outcomes of an engineering effort..... 21

4.5 Stakeholder concerns and architectural aspects..... 21

4.5.1 Stakeholders and their perspectives..... 21

4.5.2 Stakeholder concerns..... 22

4.5.3 Architectural aspects..... 22

4.6 Architecture viewpoints and architecture views..... 24

4.6.1 Architecture viewpoints..... 24

4.6.2 Architecture views..... 24

4.7 Architecture frameworks and architecture description languages..... 25

4.7.1 Architecture description framework..... 25

4.7.2 Architecture evaluation framework..... 25

4.7.3 Architecture description language..... 26

4.8 Architecture descriptions and architecture models..... 26

4.8.1 Architecture descriptions..... 26

4.8.2 Structuring architecture descriptions..... 26

4.8.3 Architecture models..... 27

4.9 Reference architectures and reference models..... 28

4.9.1 Uses of reference architectures..... 28

4.9.2 Reference models, styles and design patterns..... 29

4.9.3 Reference architecture descriptions..... 29

4.10 Life cycle and life history..... 29

4.10.1 Overview of life cycle concepts..... 29

4.10.2 Visual representation of life cycle concepts..... 30

4.10.3 Life cycle elements..... 31

4.10.4 Life cycle modelling..... 32

4.11	Architecting precepts.....	33
4.11.1	Architecture governance and management precepts.....	33
4.11.2	Architecture evaluation precepts.....	34
4.11.3	Architecture enablement precepts.....	35
5	Architecting principles.....	36
5.1	Introduction.....	36
5.2	Principles about the purpose of an architecture.....	37
5.2.1	Separation of concerns.....	37
5.2.2	Identification of key characteristics.....	38
5.2.3	Embodiment of decisions.....	39
5.2.4	Abstraction of entities.....	40
5.3	Principles about the focus of an architecture.....	41
5.3.1	Focus on value.....	41
5.3.2	Focus on effectiveness.....	42
5.3.3	Focus on intended use.....	43
5.3.4	Focus on decisions.....	44
5.4	Principles about architecting outcomes.....	45
5.4.1	Formulation of architecture features.....	45
5.4.2	Definition of relationships and interfaces.....	46
5.4.3	Development of entity realization principles.....	47
5.4.4	Development of entity evolution principles.....	48
	Annex A (informative) Notes on terms and concepts.....	50
	Annex B (informative) Determining architectural significance of concepts and properties.....	60
	Annex C (informative) A formal approach to distinguish architecture design statements from other design statements.....	68
	Bibliography.....	72

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized 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 rules given in 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 and IEC 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.

ISO/IEC/IEEE 42024 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*, in cooperation with the Software and Systems Engineering Standards Committee of the Computer Society of the IEEE, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

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

This document specifies the fundamentals of architecture practice by providing a core set of vocabulary, architecture concepts and architecting principles. The intent of this document is to help the reader understand these fundamentals and realize greater value from architecture-related standards through increased uniformity and better integration. Architecting, as a means to align form, function and fitness for purpose, is applied by organizations, projects, teams and individuals to facilitate understanding of the challenges faced by a variety of stakeholders (including downstream developers, implementors, integration authorities, certification bodies and other collaborators) and to address their concerns while also dealing with potential opportunities to be pursued.

In addition to an historic association of architecture with buildings and construction, the usage of architecture today has evolved into a commonly used practice for other sectors such as commercial, civil and defence, where the value proposition of architecture is increasingly acknowledged in various fields of application. Humankind is building exceedingly complex entities at an accelerated rate in these fields.

Organizations affected by these entities are faced with new opportunities to pursue along with an increase in architectural challenges when developing architectures for enterprises, business components, capability areas, mission areas, missions, systems, systems of systems, families of systems, infrastructure, products, services, product lines, hardware, software applications, technologies, etc. A common basis for aligning architecture concepts and architecting principles across various domains can improve the interoperability and harmonization of architecture standards and the resulting architecting work products.

To address the intimate and often misunderstood connection between architecture and design, this document clarifies the scope of architecture practice by introducing a fundamental distinction between two types of design. It considers ‘architecture design’ to entail the information that captures the strategic and enduring concepts that result from an architecting effort. This is distinct from ‘solution design’, which specifies the more detailed, tactical information necessary for implementation by subsequent engineering efforts. This conceptual separation, which is elaborated in [Clauses 3](#) and [4](#), is intended to provide clarity and complement existing engineering and design standards, especially for ISO/IEC/IEEE 15288 and ISO/IEC/IEEE 12207:Ed2 (FDIS).

Architecture is a topic of tremendous variety. This document approaches the topic from the perspective of the enterprise, systems and software fields of application. This document is intended for organizations and projects that are:

- developing, updating, managing, governing and maintaining architectures and architecture descriptions;
- seeking sustained success through the implementation of architecture practice;
- seeking to improve communication about the architecture and the architectural objectives through a common understanding of the foundational architecture concepts and architecting principles;
- performing conformity assessments against the requirements of architecture-related standards, specifications and organizational practices;
- establishing and maintaining integration and alignment between architectures;
- using architectures to enable better governance of the life cycles and life cycle processes of entities.

Organizations that choose to employ architecture concepts and architecting principles can expect to see improvements in their ability to create value for stakeholders. The adoption of more formalised and comprehensive architecture practice can be a strategic decision for such organizations.

A well-defined architecture delivers a range of benefits that can result in strategic, technical and organizational advantages:

- **Strategic alignment and planning:** It increases the clarity, efficiency and effectiveness of transformation and modernization initiatives by helping to define roadmaps and plans. It promotes coherence between enterprise-level goals and technical solutions (e.g. systems, hardware, software, services and processes), framing the development and usage of these solutions to meet consumer and stakeholder demand;

- Technical cohesion and evolution: It improves interoperability and compatibility between disparate systems, services and technologies. It promotes uniformity and consistency across a given domain, enabling commonalities between related entities and facilitating the reuse of solution elements. This provides a stable basis for managing evolution, offering guidelines for making changes that do not compromise stakeholder value, and for making provisions to accommodate future changes;
- Organizational enablement: It provides a basis for identifying necessary organizational and team member competencies, facilitating the on-boarding and training of new members of the organization and new customers;
- Stakeholder engagement and risk management: It provides a framework to express, share knowledge about, and evaluate challenges, opportunities and motivating factors. This fosters greater stakeholder engagement and cooperation in decision-making and provides a basis for the proactive identification of risks and opportunities.

This document is informed by existing architecture-related standards as illustrated in [Figure 1](#).

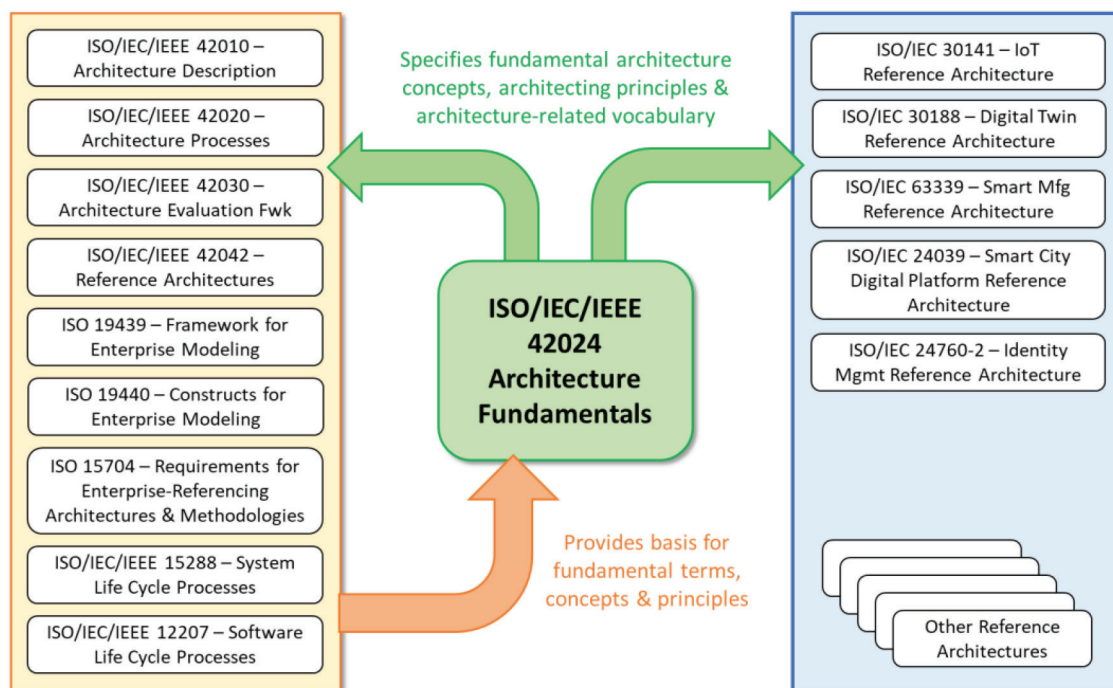


Figure 1 — Architecture-related standards

The business community benefits from using this document since it specifies architecture concepts and architecting principles, so they are better utilized across the organization. As a result, architecture-related decisions can align better with the overarching goals and strategies of the business and can facilitate increased agility in architecting new and improved enterprises, products and services.

The architecting community benefits from using this document since it provides a common foundation to communicate and share ideas about the attributes of a more mature and effective architecture practice. This document can help architecting efforts to identify and define the relevant challenges and opportunities, and more effectively convey the fundamental concepts and essential properties of an entity and communicate their recommendations more clearly to stakeholders.

The professional community of standards and reference architecture development organizations benefit from this document since it specifies architecture concepts, architecting principles and the associated vocabulary for use by these organizations. This leads to a better understanding of architecture fundamentals within the professional community and contributes to more effective standards and reference architectures as a result.

The educational community benefits from using this document since it establishes core architecture concepts and architecting principles, which can then be disseminated, built upon, and refined over time. This document helps with professional practice and its development since practitioners can more readily apply documented best practices and stay more current with architecture trends and developments.

Enterprise, systems and software — Architecture fundamentals

1 Scope

This document specifies a core set of vocabulary, architecture concepts and architecting principles associated with the architecture practice applicable to various entities such as enterprises, business components, capability areas, mission areas, missions, systems, systems of systems, families of systems, infrastructure, products, services, product lines, hardware, software applications and technologies.

This document is intended for:

- participants in architecting efforts;
- developers of architectures, architecture descriptions, reference architectures, reference architecture descriptions and reference models;
- stakeholders using architectures as the baseline for their activities and in understanding how and to what extent their concerns have been or will be satisfied;
- providers of architecture-related standards, guidelines, training, education and evaluation instruments used for assessment of architecture practices and architecting work products;
- developers of architecture description languages, architecture modelling languages, architecture frameworks, architecture development and modelling methodologies, and architecture-related tools and technologies.

This document does not address terms and definitions unique to specific application domains or areas. This document does not address concepts related to architecture competencies.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

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

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

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

NOTE 1 For additional terms and definitions in the field of systems and software engineering, see ISO/IEC/IEEE 24765, which is published periodically as a “snapshot” of the SEVOCAB (Systems and software Engineering Vocabulary) database and is publicly accessible at www.computer.org/sevocab.

NOTE 2 The terms in this clause are listed in systematic order since this is the preferred ordering scheme according to ISO/IEC Directives, Part 2, 16.4, Principles and rules for the structure and draft of ISO and IEC documents. According to this directive, “Alphabetic order is the least preferred order.” The Index of Terms is provided in [3.1](#) to facilitate finding the definitions for terms of interest.

Terms and definitions for this document are organized according to the following categories:

- architecting concepts;
- entity related concepts;
- architecture design concepts;
- architecture description concepts;
- entity life cycle concepts.

3.1 Index of terms

The purpose of this Index is to facilitate access to the terminological entries in this clause.

architecting	3.2.3	information package, architecture	3.4.3
architecting principle	3.2.4	information package, solution	3.4.6
architecture	3.2.1	life cycle	3.6.1
architecture, reference	3.2.7	life cycle phase	3.6.2
architecture description	3.5.1	life cycle stage	3.6.3
architecture design	3.4.2	life history	3.6.4
architecture feature	3.2.2	metamodel	3.5.6
architecture framework	3.2.6	model	3.5.5
architecture information package	3.4.3	model, reference	3.5.7
architecture principle	3.2.5	perspective, stakeholder	3.4.8
architecture view	3.5.3	phase, life cycle	3.6.2
architecture viewpoint	3.5.4	principle, architecting	3.2.4
aspect	3.3.7	principle, architecture	3.2.5
concern	3.4.9	reference architecture	3.2.7
context	3.3.4	reference model	3.5.7
description, architecture	3.5.1	solution description	3.5.2
description, solution	3.5.2	solution design	3.4.5
design	3.4.1	solution entity	3.3.2
design, architecture	3.4.2	solution information package	3.4.6
design, solution	3.4.5	stage, life cycle	3.6.3
dimension	3.3.8	stakeholder	3.4.7

engineering effort	3.4.4	stakeholder perspective	3.4.8
enterprise	3.3.5	system	3.3.6
entity, solution	3.3.2	view, architecture	3.5.3
entity of interest	3.3.1	viewpoint, architecture	3.5.4
environment	3.3.3	information package, architecture	3.4.3
framework, architecture	3.2.6	information package, solution	3.4.6

3.2 Architecting concepts

3.2.1

architecture

fundamental concepts and essential properties that manifest in an entity in its *environment* ([3.3.3](#))

Note 1 to entry: Fundamental concepts include decisions, principles or patterns that shape the structure, operation and long-term viability of an entity. Essential properties include attributes, characteristics or qualities that sometimes are compounded from those of constituent elements, latent or emergent in nature, which are critical to stakeholder value and arise from strategic decisions that are especially difficult to change at a later time.

Note 2 to entry: Every entity has an “architecture”, whether that architecture is described or not. Architecture is characteristic of its identity. The description of that architecture is based on insight into its structure, the relationships between the components that make up those structures, their behaviours, and the (intended or perceived) value and purpose.

Note 3 to entry: The fundamental concepts and essential properties are manifest in an entity in the sense of “making an appearance” in the entity when it is realized. In other words, the entity is a manifestation of these concepts and properties.

3.2.2

architecture feature

fundamental concept or essential property that is value-focused, holistic and enduring

Note 1 to entry: Fundamental concepts include decisions, principles or patterns that shape the structure, operation and long-term viability of an entity. Essential properties include attributes, characteristics or qualities that sometimes are compounded from those of constituent elements, latent or emergent in nature, which are critical to stakeholder value and arise from strategic decisions that may be difficult to change at a later time.

Note 2 to entry: ‘Essential’ and ‘fundamental’ are terms often used interchangeably to describe if something is necessary or crucial. However, there is a subtle difference between the two. A property is ‘essential’ if it is absolutely necessary or indispensable, or if it is crucial for the existence or functioning of the entity, while a concept is ‘fundamental’ if it is foundational or basic to the understanding of the purpose or the nature of an entity. A fundamental concept can also be used as the basis for elaborating the concept or for deriving new concepts.

Note 3 to entry: Architecture features of a component of an entity are not necessarily architecture features of the entity itself. An architecture feature of the entity applies to every component of the entity, i.e. the architecture of the entity constrains the architecture of its components.

3.2.3

architecting

conceiving, defining, analysing, expressing, documenting, communicating, certifying proper implementation of, extracting, applying, maintaining or improving an *architecture* ([3.2.1](#))

[SOURCE: ISO/IEC/IEEE 42010:2022, modified – Removed ‘for an entity’, removed note to entry.]

3.2.4**architecting principle**

fundamental belief, norm, rule or value governing or guiding an *architecting* (3.2.3) effort that holds regardless of the entity being architected or technique being used

Note 1 to entry: Architecting principles are those that apply to architecting efforts. Some of the key architecting principles that are fundamental to the architecture practice are discussed in [Clause 5](#).

3.2.5**architecture principle**

fundamental belief, norm, rule or value about an architecture that influences the realization of the entity, its *life cycle* (3.6.1) and its *context* (3.3.4)

Note 1 to entry: Architecture principles can be used to help identify architecture features to be included in the architecture design. See the “architecting” principles at 5.4.3 and 5.4.4 on the development of “architecture” principles that guide the realization and evolution of an entity, respectively.

3.2.6**architecture framework**

conventions, principles and practices for architecture-related activities established within a domain of activity or within a community of *stakeholders* (3.4.7)

EXAMPLE Unified Architecture Framework (UAF), Generalised Enterprise Reference Architecture and Methodology (GERAM) which includes an architecture description framework (called the GERA Modelling Framework), Reference Model for Open Distributed Processing (RM-ODP).

3.2.7**reference architecture****RA**

architecture (3.2.1) that constitutes a standard or common approach to a problem, circumstance or opportunity within a field of application, domain of activity or area of concern

Note 1 to entry: A reference architecture description can serve as a standard or outline comprising general or common concepts, properties and principles that apply to a category of entities in a targeted field, domain or area.

Note 2 to entry: A reference architecture description can include or refer to reference models that provide a general or common approach to a recurring problem or circumstance.

Note 3 to entry: A reference architecture typically applies to a category of entities and is developed for fields such as artificial intelligence (AI), internet of things (IoT), cloud computing, cybersecurity, big data, smart cities, smart manufacturing, smart farming, agriculture, digital twin, telecommunications, aerospace, defence, banking, finance, insurance, energy, automotive, shipping, hospitality, healthcare, supply chain, transportation, manufacturing, production and infrastructure.

3.3 Entity related concepts**3.3.1****entity of interest****EOI**

entity examined, created, changed or otherwise influenced by *architecting* (3.2.3) and engineering efforts

Note 1 to entry: An entity is ‘anything perceivable or conceivable’ [ISO 9000:2015]. Examples of entities provided in ISO 9000:2015 are: product, service, process, person, organization, system or resource. In this document, entities include enterprises, business components, capability areas, mission areas, missions, systems, systems of systems, families of systems, infrastructure, products, services, product lines, hardware items, software applications and technologies, among other things.

Note 2 to entry: The terms ‘entity’ and ‘object’ are catch-all terms analogous to ‘something’. In terms of terminology, ‘object’ is commonly used in this way. In terms of ontology, ‘entity’ and ‘thing’ are commonly used. Objects can be material (e.g. an engine, a sheet of paper, a diamond), non-material (e.g. conversion ratio, a project plan, repair procedure) or imagined (e.g. an organization in a future state, a unicorn, a scientific theory).

3.3.2**solution entity**

entity of interest (3.3.1) and other entities necessary to sustain the entity of interest in one or more *context* (3.3.4) over its *life cycle* (3.6.1) that is a realization of the architectural entities of interest

Note 1 to entry: Sustainment of the solution entity involves dealing with several properties, such as continued viability, adaptability, evolvability, affordability, operational assurance and logistics support.

3.3.3**environment**

totality of external entities, situations, conditions and circumstances enveloping an entity for a particular domain of interest

Note 1 to entry: An entity can have multiple environments, each one associated with a particular solution or portion of a solution that the entity is part of, or associated with different domains such as technological, economic, political, social, legal and ecological domains. See 4.3.5 for discussion about environment and context concepts.

3.3.4**context**

set of entities, situations, conditions and circumstances within the environment that shape, influence, relate to, or interact with an entity

Note 1 to entry: Context can help determine, specify or clarify the meaning of an event or other occurrence with respect to an entity.

Note 2 to entry: The relations between an entity and things in the context can involve or enable interactions with that entity. These interactions can include exchanges of information, material, energy, data, etc.

3.3.5**enterprise**

human undertaking or venture for dealing with a complex and challenging situation or circumstance

Note 1 to entry: An enterprise in its early stages usually has implicit or possibly ill-formed strategy, goals and objectives. A mature enterprise usually has explicit and clearly defined strategy, goals and objectives.

Note 2 to entry: It is common for an enterprise to be equated with an organization. However, in this document, the enterprise term is used to indicate the complex and challenging activity rather than the organization (or set of organizations) that participates in such an undertaking or venture.

Note 3 to entry: In this document, an enterprise can be an entity of interest that is subject to the architecting effort. An architecting effort can help clarify and make explicit the strategy, goals and objectives for an undertaking or venture. Architecting can help change the enterprise specification and scope over time as the need arises.

3.3.6**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 complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, services and personnel required for operations and support to the extent necessary for self-sufficient use in its intended environment.

Note 2 to entry: In this document, a system can be an entity of interest that is subject to the architecting effort.

[SOURCE: ISO/IEC/IEEE 15288:2023, modified – Removed notes 1 and 2 to entry, added note 2 to entry.]

3.3.7**aspect**

part of an entity's character or nature

Note 1 to entry: Aspects govern the selection of a subset of properties of an entity, represented as entity attributes,^[34] which enables a model based on that aspect to answer a defined set of questions about the entity in its context. Aspects can be mapped to an architectural dimension to help identify and formulate the appropriate questions regarding the variety of concerns about an entity.

Note 2 to entry: Aspects are characterized by a collection of observations, statements of fact or descriptive models about an entity feature or group of related features.

3.3.8 dimension

coherent collection of *aspects* (3.3.7) relevant to a domain or a particular field of knowledge or discipline

Note 1 to entry: A dimension is a set of aspects of the same type, residing in the same space.^[57] Coherence among aspects within the collection refers to a shared relationship type, a meta-property of each aspect similar to membership in a set of a shared type. Each aspect in a dimension is differentiated within that shared type by some distinguishing feature such as the phases (aspects) of a life cycle (dimension).

Note 2 to entry: The coherence requirement of the dimension can result in a collection of aspects that are unordered, partially ordered, fully ordered, or related in some other manner, or not ordered in any way.

[SOURCE: IEC 63339:2024, modified – Changed ‘manufacturing domain’ to ‘domain’, expanded note 1 to entry, added note 2 to entry.]

3.4 Architecture design concepts

3.4.1 design, noun

information necessary and sufficient for creation, transformation or understanding of an entity

Note 1 to entry: Design as used in ISO/IEC/IEEE 15288:2023, 3.13 means the “specification of system elements and their relationships, that is sufficiently complete to support compliant implementation of the architecture” and that usage is a special case of the term ‘solution design’ as used in this document (see 4.4.2). Architecture design produced by the architecting effort is typically upstream from the engineering effort that produces the solution design but sometimes this happens concurrently.

Note 2 to entry: Often, the term ‘system design’ reflects the combination of work products resulting from all architecting and engineering efforts associated with that system. The same is true for software design and enterprise design cases.

3.4.2 architecture design, noun

information necessary and sufficient for creation, transformation or understanding of an *architecture* (3.2.1)

Note 1 to entry: In this document ‘architecture design’ refers to an output of the architecting effort, not to the effort itself (see 4.2.4). Architecture design includes information for subsequent elaboration in a solution design.

Note 2 to entry: Architecture design can include architecture descriptions and can help create necessary information to support engineering efforts and other related efforts such as marketing, business development, finance, manufacturing, test, integration, deployment, logistics and operations.

Note 3 to entry: Architecture design can include an architecture information package (see 4.4.1).

3.4.3 architecture information package

work product of an *architecting* (3.2.3) effort that conveys information about the *architecture design* (3.4.2) to be used as the basis for engineering and other architecture-related efforts

Note 1 to entry: The architecture information package term is used when the architecting efforts are intended to elaborate the architecture of an entity, The solution information package term (see 3.4.6) is used when the engineering efforts are intended to produce a solution design that provides the basis for realization of a solution entity that is aligned with the architecture.

Note 2 to entry: The architecture information package informs development of one or more architecture descriptions and can include additional information necessary to support an engineering effort and other related efforts such as marketing, business development, finance, manufacturing, test, integration, deployment, logistics and operations. See 4.4.1 for typical contents of this package.

3.4.4**engineering effort**

transformation of an *architecture design* (3.4.2) for a *solution entity* (3.3.2) into a *solution design* (3.4.5) by elaborating, refining, expanding and detailing to the extent that the design is sufficiently complete for implementation

Note 1 to entry: In this document 'solution design' refers to the output of the engineering effort rather than to the activity that produces this output.

Note 2 to entry: Architecture design produced by the architecting effort is typically upstream from the engineering effort that produces the solution design but sometimes this happens concurrently.

3.4.5**solution design**

information necessary and sufficient for the implementation of *solution entities* (3.3.2) or for further *engineering efforts* (3.4.4)

Note 1 to entry: The solution design refines and expands upon the architecture design by specifying the particulars associated with each architecture design element to specify the ways and means of implementation for that element.

3.4.6**solution information package**

work product of an *engineering effort* (3.4.4) that conveys information about the *solution design* (3.4.5) to be used as the basis for implementation and other engineering-related efforts

Note 1 to entry: Solution design provides the physical structure, behaviour, temporal and spatial relationships, and other attributes of an entity and its components suitable for implementation. This package also includes additional information necessary in support of solution design definition effort and in related efforts such as marketing, business development, finance, manufacturing, test, integration, deployment, logistics and operations.

3.4.7**stakeholder**

role, position, individual, organization, or categories thereof, having a right, share, claim or other interest in an entity

Note 1 to entry: Both direct and indirect stakeholders exist. Indirect stakeholders are affected or impacted by an entity or its architecture by having intervening factors, persons or influences while direct stakeholders are engaged in the conceptualization, evaluation and elaboration of the architecture.

3.4.8**stakeholder perspective**

way of perceiving or thinking about an entity, especially related to stakeholder *concerns* (3.4.9)

EXAMPLE Various domains have customary perspectives: (1) industrial production systems – operational and financial perspectives; (2) banking systems – business, management, acquisition, supply chain perspectives; (3) mobile applications – development, deployment, customization perspectives; (4) hospitality services – provider, customer perspectives; (5) content provider platform – data user, data provider perspectives.

[SOURCE: ISO/IEC/IEEE 42010:2022, modified – Removed notes to entry, replaced examples, added 'perceiving or']

3.4.9**concern**

matter of interest, relevance or importance to or impact on a *stakeholder* (3.4.7)

Note 1 to entry: Stakeholder concerns which can affect the architecture include, for example, those that are technical in nature; organizational issues; socio-political factors; legal, ethical and regulatory considerations; and social equity issues. While some concerns are emergent manifestations of underlying causes that architecting is trying to address, new concerns can emerge as architecture decisions are made.

Note 2 to entry: Concerns can be identified with regards to an entity or independently, such as with regards to environment, scenario, situation or use case related to that entity.

EXAMPLE For a Taxi Aggregator system: (1) wait time, price, revenue options and commission are the typical concerns of the driver; (2) cost efficiency, payment options, route trace and safety are the typical concerns of the rider; (3) safety, commission, rate and exploitation are the typical concerns of the regulatory bodies. Additional examples are provided in [4.5.1](#).

[SOURCE: ISO/IEC/IEEE 42010:2022, modified – Removed notes to entry, added notes to entry, added example.]

3.5 Architecture description concepts

3.5.1

architecture description

AD

work product used to express an *architecture* ([3.2.1](#))

Note 1 to entry: The purpose for which an architecture description is created or utilized determines the scope of an architecture description for an entity. An AD is complete if it addresses the architecture with respect to its purpose. Often, several purpose-specific architecture descriptions are needed to wholly capture an entity's architecture, and even then, aspects of the entity's architecture are overlooked or ignored. See [4.4.1](#) and [B.7](#) for more information regarding wholeness of architecture.

[SOURCE: ISO/IEC/IEEE 42010:2022, modified – Removed note to entry, added note to entry.]

3.5.2

solution description

work product that expresses a *solution entity* ([3.3.2](#))

Note 1 to entry: A solution description usually covers several aspects of the solution entity over its full life cycle, possibly including objectives and requirements, manufacturing and construction considerations and plans, test and evaluation philosophy and approach, operational and employment concepts and characteristics, maintenance and sustainment approaches, deployment and distribution plans, etc.

3.5.3

architecture view

expression of an *architecture* ([3.2.1](#)) addressing *concerns* ([3.4.9](#)) and reflecting *aspects* ([3.3.7](#))

Note 1 to entry: The primary purpose of an architecture view is to convey useful and practical information to people who are involved in the architecting and engineering efforts and to other stakeholders to communicate the manner in which their concerns are addressed. Architecture views express with respect to concerns, but the concerns belong to stakeholders, as defined in [3.4.9](#). As in the stated example of [3.4.9](#), neither "revenue options" nor "payment options" would likely be a view, but instead a "financial view" is likely needed to address the revenue and payment options as expressed in the architecture.

Note 2 to entry: Architecture views often become the basis for further architecting or engineering activities.

3.5.4

architecture viewpoint

conventions for the creation, interpretation and use of an *architecture view* ([3.5.3](#))

Note 1 to entry: Architecture viewpoints are associated with stakeholder perspectives and frame their respective concerns, and they provide the conventions by which views can be created with architectural aspects in mind.

3.5.5

model

representation of an entity or collection of entities under conditions or situations of interest

Note 1 to entry: A model is typically used to help portray, specify, understand or predict the properties or characteristics of an entity or collection of entities.

3.5.6**metamodel**

model (3.5.5) that defines the elements and properties needed to create a model

Note 1 to entry: A metamodel is a model that consists of statements about terms used in models. Hence, a metamodel is also a model, but its universe of discourse is a set of models. A metamodel defines an abstract syntax and is the formal definition of concepts to be modelled and their relationships and attributes. The abstract syntax is distinguished from the concrete syntax, which is the formal definition of the representation of elements of the abstract syntax in a notation that is usually graphical or textual in nature. Rules of interpretation are associated with the abstract syntax to provide the semantics of the modelling language.

Note 2 to entry: The semantics of the terms defined in a metamodel are only defined in a limited way, because metamodels rely on denotations of the terminal symbols, as well as on implied but not explicitly defined axioms that are needed to correctly interpret models created using the terminology. Metamodels are a limited form of expression for the ontology of a field.

[SOURCE: ISO/IEC 19506:2012, modified - Removed 'special kind of', added notes to entry.]

3.5.7**reference model**

model (3.5.5) that provides a standard or common approach to a recurring problem or circumstance

Note 1 to entry: A reference architecture can rely upon a reference model that has already defined the relevant architecture features for the architecture to be derived from the reference architecture. A reference architecture description can reference or contain one or more reference models.

3.6 Entity life cycle concepts**3.6.1****life cycle**

set of *phases* (3.6.2) or *stages* (3.6.3) that an entity goes through from its conception until it ceases to exist

Note 1 to entry: The entity life cycle starts with the identification of a need for the entity and ends when it is no longer needed.

[SOURCE: ISO 15704:2019, modified – Removed notes to entry, added note to entry.]

3.6.2**life cycle phase**

focused set of activities at an extent of abstraction, undertaken to develop and evolve an entity in its *life cycle* (3.6.1)

Note 1 to entry: A phase frames a typical set of activities that act upon an entity when considered with respect to its life cycle. Actual phases are instances of these typical phases when they are implemented in time, often repeatedly.

3.6.3**life cycle stage**

period within the *life cycle* (3.6.1) of an entity that relates to the state of its evolution

Note 1 to entry: As used in this document, stages relate to major progress and achievement milestones of the entity through its life cycle. Analogous to phases, stages have a type (e.g. 'factory refurbishment stage'), and instances (the '1995 refurbishment of the factory', the '2007 refurbishment of the factory'). An actual stage (i.e. an instance of a stage type) consists of life cycle activity instances (i.e. actual phases), such as a set of completed architecture process activities, with the objective of achieving a given milestone.

3.6.4**life history**

sequence of actual *life cycle phases* (3.6.2) that an entity goes through during its lifetime

Note 1 to entry: Actual phases can be aggregated into actual stages. Each of these can, and typically do, overlap. See [A.3](#) for examples.

Note 2 to entry: There is only one 'past history' but there can be multiple possible 'future histories' (i.e. futures).

[SOURCE: ISO 15704:2019, modified – Deleted 'as steps within phases', added note to entry.]

4 Architecture concepts

4.1 Overview

This clause presents concepts that are fundamental to the architecture practice. [Clause 3](#) provides a definition for each key concept with regard to how it is used in this document. [Annex A](#) provides additional information on these terms and concepts beyond that provided in this clause.

4.2 Architecture and architecting

4.2.1 Introduction

Architecture is comprised of the fundamental concepts and essential properties that manifest in an entity in its environment. One or more entities can help attain the goals and objectives of an enterprise, organization, project or other initiative through their ability to cause or lead to desired effects and outcomes. These entities can influence and be influenced by other entities. The fundamental concepts and essential properties are conceptualized and defined through an architecting effort enabled by an understanding of the entities that can be realized as solutions that can achieve the desired effects.

Architecting involves the conceiving, defining, analysing, expressing, documenting, communicating, certifying proper implementation of, extracting, applying, maintaining, or improving an architecture for an entity. Processes associated with architecting such as conceptualizing, evaluating, elaborating, managing and governing an architecture are defined in ISO/IEC/IEEE 42020. Architecting produces information item artefacts as part of performing these processes.

NOTE Certifying the proper implementation of an architecture can be captured as an explicit statement by the architect to the client or user that the entity, as built, or otherwise realized, meets the criteria as ready for use.

Architecture applies to entities in every domain and to different kinds of considerations. Potential influences on an architecture, some of which can be embodied in the architecture, include fundamental practices, patterns, standards, regulations, policies and other driving factors. An architecture is more or less amenable to having the concepts and properties that help achieve certain desired effects, while being either focused on one consideration or balanced across a whole range of considerations.

Architecture embodies decisions and incorporates concepts and properties regarding the overall structure and behaviour of the entity, roadmapping imperatives, capability objectives, desired effects, enterprise vision, organizational directives, statutory regulations, etc. Often architecture informs the organizational decision process, particularly with respect to risk and opportunity management.

An architecture can address entities within a domain, or with respect to a perspective or concern, employing a corresponding qualifier, e.g. system architecture, business architecture.

EXAMPLE 1 (1) The 'Architecture of a Hospital Information System (HIS) entity' versus the 'Security Architecture of an HIS entity' where the latter is limited to the architecture of the security feature of the HIS from the perspective of HIS security engineers, and of hospital workers and managers. (2) The 'Architecture of the business design of the administrative operations of a hospital' versus the 'Architecture of the operating hospital' where the former is concerned with a particular subset of the hospital and the latter is concerned with all matters related to the structure and behaviour of the business of medical care provided by the hospital.

An architecture can focus on a particular subset of fundamental concepts or essential properties that are manifest in an entity, in which case a suitable qualifier is employed, e.g. information architecture, security architecture. Many different ways of qualifying an architecture exist.

EXAMPLE 2 An information architecture focuses on the information concepts or properties and simplifies complex information for users as a result; on the other hand, a security architecture focuses on the security concepts or properties and provides threat defence and mitigation measures as a result.

4.2.2 Architecting effort

The architecting effort tends to take a broader perspective of concerns than the subsequent engineering effort does.

Architecting can be applied to various entities, such as enterprises, business components, capability areas, mission areas, missions, systems of systems, families of systems, infrastructure, products, services, product lines, hardware, software applications and technologies, etc. Each entity can be understood through the mindset that matches its nature, for example an enterprise mindset for enterprise systems, a capability mindset for capability areas, a service mindset for services, a security mindset for security architecture, etc.

Architecting identifies relevant stakeholders and develops an architecture that satisfies the stakeholder needs and concerns together with related requirements. Architecting has a special focus on value delivered to stakeholders whereas engineering tends to focus on meeting established requirements. Engineering subsequent to architecting focuses on detailing the solution through defining the solution characteristics so as to yield stakeholder value. However, value to stakeholders can arise from ensuring commonalities across multiple solutions, either because they belong to a product line for which common assets can be created, or because stakeholders derive value from reusing assets across solutions.

Stakeholder value can also arise from the immediate realization of a solution but typically also is derived over the longer timescale of the solution lifetime. Given that influences are likely to change over time, stakeholder concerns are likely to relate to solution extensibility, modifiability and scalability, as well as the stability of suppliers, thereby ensuring that the solution is adaptable to anticipated changes (and unanticipated changes, to the extent possible), as well as ensuring that the downstream engineering decisions do not adversely affect these aspects of the architecture.

As illustrated in [Figure 2](#), the architecting effort identifies and addresses the relevant challenges that motivate the opportunities suitable for pursuing, which produce the desired effects (whilst also avoiding the consequences of undesired effects). Architecture drivers motivate changes to the architecture by their influence on the challenges, opportunities and desired effects to be addressed in the problem space by the architecting effort.

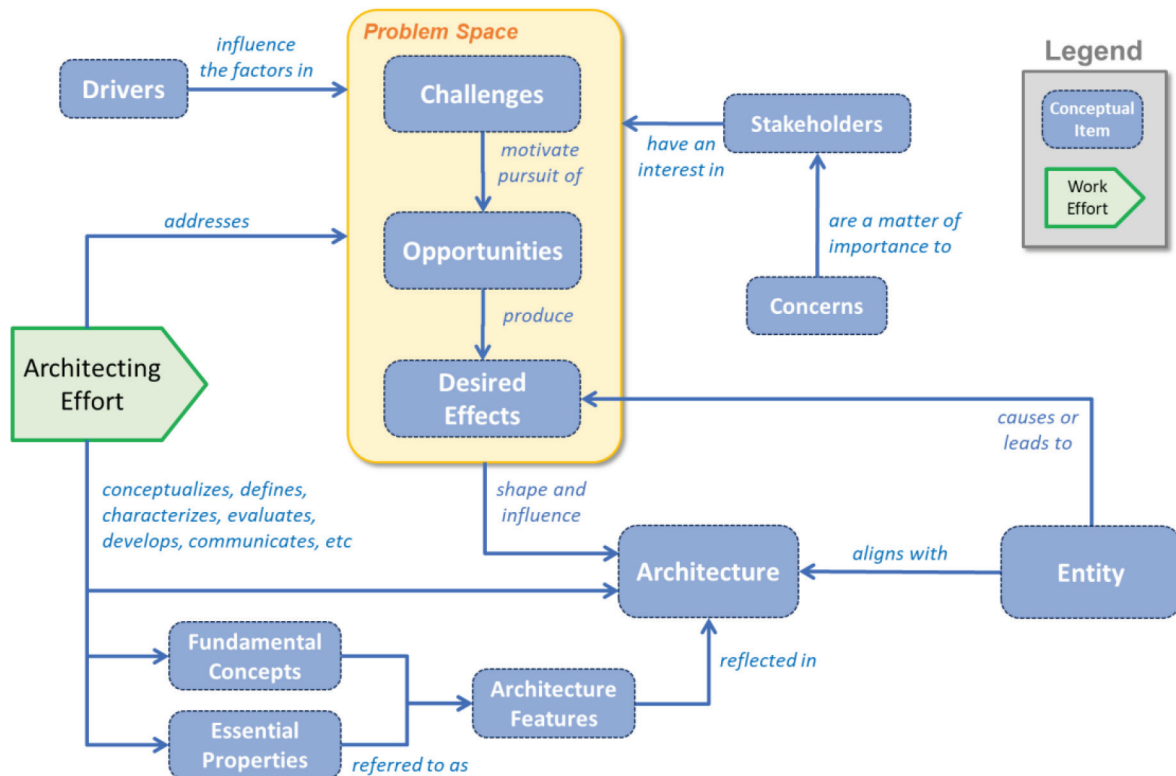


Figure 2 — Architecture motivated by challenges, opportunities and desired effects

NOTE 1 This figure is a conceptual model, not a process flow. The boxes represent concepts, and the lines represent relationships between the concepts. This is similar to a traditional entity-relationship (ER) diagram.

The challenges encountered in delivering the architecture are useful for identifying potential benefits or to identify causal and other factors. Specific influencing factors exist like profitability, competitiveness, readiness, maturity, etc. These factors in turn influence the properties of the entity itself. Architecting addresses emergent properties, whether desired and intended or unwanted and unexpected. Architecting is applicable at different, but related, levels of abstraction, highlighting the relevant architecture features that are necessary for the decisions at each level.

The desired effects shape and influence the architecture development. The failure to achieve the desired effects in the current state of affairs (i.e. achieving 'undesired' effects) can generate additional concerns on the part of stakeholders. The architecture reflects the architecture features identified and established during the architecting effort.

It should be noted that [Figure 2](#) illustrates the typical situation encountered, namely forward architecting driven by a problem requiring solution. Architecting can also be applied in a reverse architecting situation where either an existing or an already defined solution is examined to extract the architecture features.

Potential opportunities to pursue are identified that can address the pertinent challenges. By pursuing these opportunities, the architecting effort determines the fundamental concepts and essential properties that will cause or lead to achievement of the desired effects. Based on these concepts and properties, the architecting effort incorporates architecture principles and decisions into the architecture.

In summary, the architecting effort includes the following activities with respect to the key concepts portrayed in [Figure 2](#):

- 1) Determine strategic drivers and associated challenges that the architecture is intended to address.
- 2) Determine possible benefits to obtain and possible losses to avoid.
- 3) Explore opportunities to pursue that address the challenges and drivers.
- 4) Determine desired effects the architecture can be projected to achieve during various stages in the entity life cycle.
- 5) Identify candidate solution entities that can contribute to these desired effects.
- 6) Define one or more architectures that capture the intended concepts and properties and that satisfy the concerns of the stakeholders.

NOTE 2 This list is not intended to be an exhaustive list of activities involved in architecting since that is the purpose of ISO/IEC/IEEE 42020. Rather this list is intended to highlight the activities related to the key concepts in [Figure 2](#).

4.2.3 Architecture features

Architecture features are the fundamental concepts and essential properties that are value-focused, holistic and enduring. Architecture features are of vital importance for comprehension and eventual implementation of the entity. The architecture features of an entity, as specified in an architecture design (see [4.2.4](#)), possess the following qualities:

- fundamental and essential to the entity, and together constitute its architecture;
- enduring, i.e. persistent and pervasive, thereby impacting the whole of the entity over its lifetime;
- primary contributors to enabling the benefit and value the entity provides;
- applicable to the full extent of the entity's form, function and effect.

EXAMPLE Architecture features can be identified and understood by examining various aspects of an entity. For safety as a concern, for example, architecture features include fail-safe operations, physical isolation and user warnings. Furthermore, for resilience as a concern, architecture features include avoiding, withstanding and recovering from adversity. For reliability as a concern, architecture features include redundancy, life testing, real-time in situ repair. For information assurance as a concern, architecture features include access control, encryption and redundant storage. See [4.5](#) for discussion about concerns and aspects.

These features serve to identify constituent elements that, when established, enable downstream activities to address the upstream challenges, opportunities and concerns, which kicked off the architecting effort as part of, for example, a change or transformation programme, system of systems integration initiative, portfolio refresh activity, or system development project. In each case, what is fundamental or essential is dependent on the anticipated desired effects the architecture can bring about. Ultimately, the architect determines, often through experience and judgement, what is architecturally significant. Substantial change in the architecture features of an entity results in a new distinct entity.

4.2.4 Fundamental concepts and essential properties

Fundamental concepts in the architecture are the abstract or general ideas inferred or derived from specific decisions, principles or patterns that are used to shape the structure, operation and long-term viability of an entity in its environment. These concepts can be used as the basis for identification of guiding principles and for making decisions that shape the structure, operation and long-term viability of an entity. They influence the essential properties in the architecture and ensure that the entity remains adaptable and sustainable over time. They span over multiple aspects, affecting interactions, dependencies and strategic decisions. They are influenced by both internal and external factors. The selection and prioritization of these concepts are critical since they determine the entity's ability to adapt, grow and evolve over time.

Fundamental concepts in the architecture:

- clarify purpose and value;
- provide a consistent vocabulary and common language;
- establish the basic building blocks of the entity;
- help in making rational and justifiable architecture decisions;
- enable consistency and comparability.

Essential properties in the architecture are the attributes, qualities or characteristics that determine the intended or observed nature of an entity in its environment. They are compounded from those of constituent elements, latent or emergent in nature, which are critical to stakeholder value and arise from architectural decisions that are difficult to change at a later time. These properties are the general characteristics of an entity expressible in terms associated with the architecture design's inclusion of specific behaviour, structure, design, implementation, etc. They are also related to the motivating factors influencing architectural decisions and to the architecture concepts and governing principles that influence an entity's implementation.

Essential properties influence the extent to which the architecture enables the development and implementation of an entity that delivers value to its stakeholders and enables the solution entity to cause or lead to the achievement of the desired effects determined by the architecting effort (as illustrated in [Figure 3](#)).

Some of the properties of an entity are architecturally significant while others are not. In this document, the entity properties that are architecturally significant are called essential properties to distinguish them from the non-architecturally driven entity properties.

EXAMPLE 1 Examples of architecturally significant fundamental concepts are expected functionality, required performance, trustworthiness, safety, usability.

EXAMPLE 2 Examples of architecturally significant essential properties are security, reliability, maintainability, adaptability.

EXAMPLE 3 Size, weight and power are usually not essential properties but can be essential to proper functioning of the entity. The real property of concern to the architecture can be something like transportability which can then be decomposed by the engineer into relevant entity properties during the engineering effort (see [4.4.3](#)).

NOTE Aircraft weight is not necessarily an essential property of the architecture even though it is often used as an important assessment criterion for judging suitability of the architecture. This is an example of an entity property (weight) that relates to some measurable characteristics of the architecture, e.g. maximum range, optimal cruise altitude, minimum speed and capacity for the aircraft.

Essential properties in the architecture:

- establish expected measurable characteristics;
- enable clarification of delivered characteristics in terms of accomplished expectations;
- help determine the range of motivating factors to consider;
- guide selection of the architecting approach used to address particular issues;
- assist in understanding how value is delivered to stakeholders.

Entities embodying architecture can further influence, constrain or qualify these essential properties and the attendant value to different stakeholders. The extent to which the architecture addresses a combination of these properties can indicate success of the architecting effort and overall quality of the implemented entity.

4.2.5 Architecting principles and architecture principles

The set of principles related to an entity's formation and evolution over its lifetime and through interaction with its environment are an important kind of architectural concept. These principles can be externally imposed, for example through policy or doctrine, or internally imposed, for example through corporate standards or organizational methodologies. In this document, principles are considered as the fundamental beliefs, norms, rules and values that can be used as a basis for reasoning or conduct. These principles are classified into two different categories: architecting principles and architecture principles.

Architecting principles are those principles that apply to the architecting efforts and can be used for governing and guiding these efforts that hold regardless of the entity being architected or technique being used. On the other hand, architecture principles are those principles that are embedded in the architecture itself that can be used for guiding and constraining the realization and evolution of the entity. Architecture principles should be part of the architecture rationale in order to guide the development and understanding of the architecture of an entity during its lifetime, and to constrain or guide an architecture to meet an organisation's strategy, to improve coherence across a portfolio, or in the adoption of good practice.

Architecting principles are applied at various stages and phases of the architecting effort, from initial planning through to execution, review, enablement, governance and management. Consequently, these principles are more specific to the actual activities involved in the architecting efforts. That is, architecting principles govern and guide the methods, techniques and approaches used during the architecting effort. Architecting principles are about the purpose, focus and nature of architecture, and as a result are pragmatic and deal with the practical side of the architecting effort, such as stakeholder engagement, risk management, entity development and evolution, solution feasibility and suitability. A set of fundamental architecting principles are defined in [Clause 5](#).

EXAMPLE 1 Architecting principles include ideas such as: focus on value, focus on effectiveness, focus on intended use, separation of concerns, formulation of architecture features, abstraction of entities.

On the other hand, architecture principles are broad, overarching and prescriptive directions that influence how decisions should be made in a specific context and that are applied throughout the life cycle stages and phases of the entity. Consequently, these principles apply to the engineering of the solution design over its life cycle. These principles guide the development and permissible evolution of the entity during its lifetime

and are expected to influence the entity-related decisions from the concept stage to disposal through initial design, development, operation and maintenance of the entity.

EXAMPLE 2 Architecture principles include ideas such as: providing intuitive interfaces for end users, providing immediate and consistent feedback to users, aligning system behaviour with user expectations, taking advantage of high cohesion and low coupling, maximizing reuse and minimizing life cycle cost, data as an asset, modularizing to insulate each part from other parts, and ensuring the direction of dependency is in the direction of less abstraction.

Architecting principles tend to be more flexible and adaptable to multiple architecting efforts. Consequently, these principles are adjusted or refined based on specific context or changing conditions during the architecting effort. On the other hand, architecture principles tend to be rigid and long-lasting because they apply across the entire life cycle of the entity. Consequently, architecture principles provide a more stable framework for entity development, evolution, operation, governance and management decisions and alignment with the organizational directives, policies and business models.

4.3 Architecture design and solution design

4.3.1 Architecting effort to produce an architecture design

As part of an architecting effort for an entity, an architecture design aims to specify the architecture features of the entity, while the solution design aims to specify characteristics of that entity and related entities sufficient for creating those entities or otherwise bringing them into being as a realization of the designed entity.

Understanding the distinction between architecture design and solution design is important in order to enable more effective architecting results. In current practice this distinction is difficult to ascertain when there is no clear and definitive dividing line between the architecture design and the solution design concepts. That demarcation within design is often subjective and depends on the local situation, organizational factors, methodological practices, etc.

NOTE 1 As used in this document, design is either the information about (a) a plan captured from an existing entity or (b) a plan for subsequent work to perform for a new or modified entity.

NOTE 2 An architecture design is used as a basis for subsequent solution design formulation or is used for additional architecture design formulation. Architecture design produced by the architecting effort is typically upstream from the engineering effort that produces the solution design. See [A.1](#) for further discussion on the relation between architecture design, solution design and solution-related entities.

The architecting effort should address challenges or deal with difficult situations by using an architecture conceptualization approach (see ISO/IEC/IEEE 42020) that addresses both articulated stakeholder concerns and known aspects of architecture relevance (see [Figure 3](#)).

Such an approach encourages and promotes a clear understanding of the entity properties and characteristics necessary to achieve the desired effects and outcomes. The challenges or difficult situations addressed by an architecture are characterized in terms of relevant motivating factors, goals and objectives, phased deployments and roadmaps, capabilities and effects, opportunities to pursue, etc. Stakeholder concerns are usually associated with these characteristics.

Architecture design artefacts may include a wide range of specification topics and types. Design artefacts should progressively specify an entity until it is stated in a clear, consistent and sufficiently complete manner for ultimate realization.

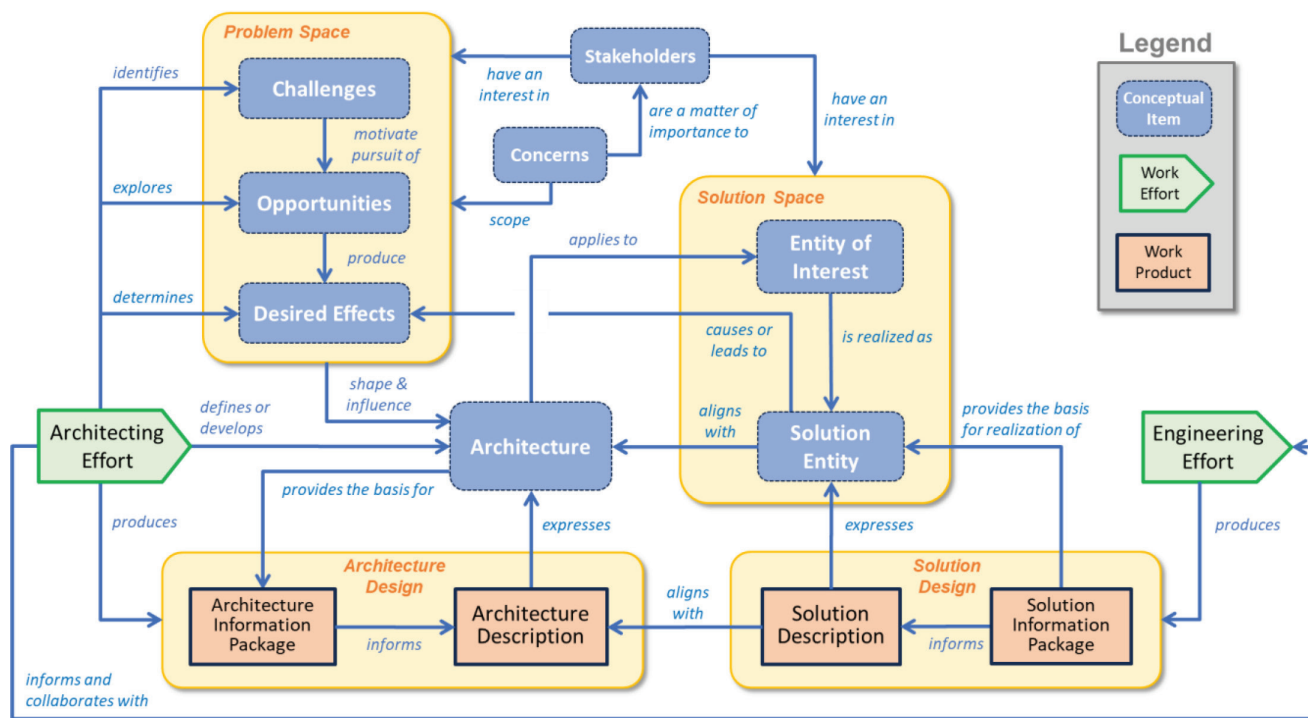


Figure 3 — Roles of architecting and engineering efforts

Other outcomes of the architecting effort result in an architecture design suitable for activities such as cost estimating and budgeting, risk identification and mitigation, project planning and scheduling, corporate governance and management, technology planning and assessment, lower-level architecting efforts and reference architecture development. These outcomes are not further discussed in the remainder of this subclause since the focus here is on an architecture design suitable for the subsequent engineering efforts that result in the solution design. See [Annex B](#) for more information regarding other expected outcomes.

An architecting effort should produce the architecture description (AD) by using an architecture elaboration approach (see ISO/IEC/IEEE 42020). The AD should contain information about the architecture features of the entity that persist after its realization, even when the solution design changes over the lifetime of the entity.

NOTE 3 ISO/IEC/IEEE 42010 standardizes the contents and form of an AD. Use of AD in this document does not imply that this item is necessarily 42010 compliant.

The term “architecture feature” is used throughout this document when referring to a fundamental concept or essential property of an entity that is value-focused, holistic and enduring. These architecture features are captured in an AD and in the architecture information package, which together make up the “architecture design” referred to in this document. These architecture features should constitute the necessary features of an entity’s architecture. These architecture features are based upon governing principles for the realization and evolution of the entity and its related life cycle processes. See [4.2](#) for more explanation on the nature of an architecture.

Architects and engineers should use the approach specified in [Annex B](#) to determine the difference between architecture design statements and solution design statements and to identify architecture features of the entity as distinct from the other non-architectural features an entity exhibits or possesses. The architecture features of an entity, as expressed in an architecture design, are:

- fundamental and essential to the entity, and together constitute its architecture;
- enduring, i.e. persistent and pervasive, thereby impacting the whole of the entity over its lifetime;
- primary contributors to enabling the benefit and value the entity provides;
- applicable to the full extent of the entity’s form, function and effect.