Title	**ISO/IEC WD1 42020 – Architecture processes**
Source	**ISO/IEC JTC 1/SC 7/WG 42 Secretariat**
Reference	**ISO/IEC JTC 1/SC 7/WG 42 N0137**
Status	**Final (v0.8, 2015-08-31)**
Replaces	**N/A**
Document type	**Working Draft (WD)**
Action ID	**FYI and ACT**
Due date	For WG schedule, comments by 2015-10-28 are appreciated.

Short description

This document is the first working draft of the intended International standard ISO/IEC 42020 Architecture processes. The project was approved by a NWIP ballot that contained an annotated outline called WD0 (zero).

The plan is to have ISO/IEC JTC1/SC7 technical experts provide comments on WD1 by 2015-10-28, to be dispositioned by WG 42 at its next work meeting in Hoboken, NJ, USA, the following week. The project schedule suggests that a WD2 be subsequently issued for commenting in Q1 2016.

When providing comments, please use the dedicated Excel spread-sheet template provided (42N0138). The use of line numbers for document reference (navigation) when providing comments is mandatory, whereas references to page numbers, clauses and paragraphs are optional.

ISO/IEC WD1 42020:201x(E)

Edition: JTC1/SC7/WG42

Date: *31/08/2015*

Version: ***0.8***

Secretariat: JTC1/SC7

Systems and software engineering — Architecture processes

Ingénierie Système et Logiciel — Processus liés à l'architecture

Warning

This document is an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

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.

EDITORIAL NOTE: Editorial notes, like this one, are inserted at various points in the draft to capture temporary ideas and provide guidance or other information for the reviewers. These editorial notes will be removed in the final released version of the standard. Notes labelled merely as "NOTE" will remain within the standard after publication to serve as guidance and provide informative content for the user of the standard

Contents

Page

1	Overview	1
1.1	Scope	1
1.2	Purpose	1
1.3	Field of application	2
2	Conformance	3
2.1	Intended usage	3
2.2	Full conformance	4
2.3	Tailored conformance	5
3	Normative references	5
4	Terms and definitions	5
5	Key concepts and application of this International Standard	11
5.1	Introduction	11
5.2	Contexts of architecture activities	12
5.3	Architecture and life concepts	12
5.4	Object of the architecture	12
5.5	Architecture kinds	12
5.6	Architecture and architecture description	13
5.7	Process concepts	13
5.8	Process application	14
5.9	Process reference model	15
5.10	Processes in this standard	16
6	Processes	17
6.1	Architecture governance process	18
6.2	Architecture management process	21
6.3	Architecture conceptualisation process	25
6.4	Architecture evaluation process	28
6.5	Architecture elaboration process	32
6.6	Architecture enablement process	34
7	External interactions	37
7.1	Relationship with system life cycle processes	37
7.2	Relationship with enterprise processes	38
8	Architecture-related work products	39
Annex A	(normative) Tailoring process	40
A.1	Introduction	40
A.2	Tailoring process steps	40
Annex B	(normative) Interactions with system life cycle processes	42
Annex C	(informative) Relationship with other standards	46
Annex D	(informative) Notes on terms and concepts	48
D.1	Architecting scenarios and approaches	48
D.2	Architecting kinds and styles	50
D.3	Architecture motivation model	54
D.4	Architecture quality attributes	54
Annex E	(informative) Architecture process-enabling resources	56

Figures

Page

Figure 1: Architecture process interactions 16

Figure 2: Key information flows between the architecting processes 17

Figure 3 : Architecture use along a system life cycle 37

Figure 4 : Architecture use along an enterprise life cycle 38

Figure 5: Main relationships between ISO/IEC 42020 and other ISO standards 47

Tables

Page

Table 1: Architecture processes 17

Table 2: Uses of architecture by 15288 processes 42

Table 3: Information used by 42020 architecture processes 44

Table Annex D-1: Quality attributes and sub-attributes from ISO 25010 55

Foreword

ISO (the International Organisation for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organisations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

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.

ISO/IEC WD1 42020 was prepared by Joint Technical Committee ISO/JTC 1, Information Technology, Subcommittee SC 7, *Systems and software engineering*.

Introduction

The complexity of man-made systems has grown to an unprecedented level. This has led to new opportunities, but also to increased challenges, for the organizations that create and utilize systems. Concepts, principles and procedures of architecting are increasingly applied to help manage the complexity faced by stakeholders of systems. Architecture processes are performed in various organisations and for many reasons. Today, usage of architecture frameworks has evolved into a state-of-the-art practice for both civilian and military domains. Architecture activity is now considered as strategic in both projects and enterprises.

The value-proposition of architecture is increasingly acknowledged within enterprises and the engineering disciplines. This standard complements the architecture-related processes of ISO/IEC/IEEE 15288, ISO/IEC/IEEE 12207 and ISO 15704 with a set of requirements enabling architects to more effectively implement architecture practices.

The benefit for enterprises will be to enable more efficient and effective implementation of architecture processes that ensures greater impact of the architecture on enterprise success.

Standardized architecture activities will benefit the architecting professional practice:

- A process framework contributing to the identification of job roles in the organisation, along with requisite skills and competencies.
- Standardized architecture approaches adopted by enterprise, system, information technology, product and service architects.

The following stakeholder categories will benefit from the proposed standard:

- Solution acquirers to formalize the business context, evaluate providers' proposals, identify alternatives, and make informed decisions;
- Solution providers to understand the problem/request, elaborate a proposal in their solution space, and define and justify their deliveries;
- Solution users to express the operational context, formalize their needs, and evaluate providers' proposals in their solution space;
- Decision makers to use architectures as a source of information and rationale for the decisions to be made;
- Other bodies such as legal, safety and security authorities, to assess compliance with standards, policies, directives, treaties, regulation, and laws.

This standard specifies processes and work products related to architecture. This description is complemented by relationships with the systems life cycle and enterprise processes.

The purpose of the standard is to provide requirements on the governance, management, conceptualisation, evaluation and elaboration of architectures.

Systems and software engineering — Architecture processes

1 Overview

1.1 Scope

This International Standard establishes a common framework of process descriptions for the development and use of architectures.

This International Standard defines a set of processes and associated terminology for the architecting, managing, and governing of architectures, along with enabling activities in support of the architecting processes.

These processes can be applied to either a problem situation or an opportunity pursuit situation. These processes can be applied in a project, as well as in the enterprise within which multiple projects will be performed. Selected sets of these processes can be applied throughout an architecture's life cycle for managing and performing within the stages involved in the development and use of the architecture. This is accomplished through the involvement of all stakeholders, with the ultimate goal of achieving stakeholder satisfaction.

Organizations and projects can use these processes when acquiring and supplying systems, products and services. They can use these processes when architecting an enterprise, a system of systems, a collection of systems, a class of systems, a family of systems, a product line, an individual system, a portion of a system, a product, a service, an individual hardware or software item, or any other entity that is amenable to architectural definition.

NOTE Systems addressed by this International Standard are man-made and may be configured with one or more of the following system elements: hardware, software, data, humans, processes (e.g., processes for providing service to users), procedures (e.g., operator instructions), facilities, materials and naturally occurring entities.

When the architecture is for a system element such as a software or hardware item, the processes in this International Standard may be used to define the architecture of that system element.

1.2 Purpose

The purpose of this International Standard is to provide a defined set of processes in the life cycle of an architecture or the life cycle of systems related to that architecture.

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

1.3 Field of application

1.3.1 Application items

This International Standard applies to the full life cycle of an enterprise, including formulation, startup, steady state operations, contingency operations, mergers and acquisitions, and shutdown, and to the management and control of portfolios, programs and projects within the enterprise. The life cycle processes of this International Standard can be applied concurrently, iteratively and recursively to an enterprise and incrementally to its elements.

This International Standard 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 processes of this International Standard can be applied concurrently, iteratively and recursively to a system and incrementally to its elements.

There is a wide variety of systems in terms of their purpose, domain of application, complexity, size, novelty, adaptability, quantities, locations, life spans and evolution. This International Standard describes the processes for the development and use of architectures comprised of man-made systems. It therefore applies to one-of-a-kind systems, mass-produced systems and customized, 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.

1.3.2 Process execution

This International Standard provides a process reference model characterized in terms of the process purpose and the process outcomes that result from the successful execution of the activity tasks. This International Standard can therefore be used as a reference model to support process assessment as specified in ISO/IEC 15504-2. Annex C in ISO/IEC/IEEE 15288 provides information regarding the use of the architecture processes as a process reference model. The processes in this International Standard are specified in terms of activities and tasks comprising those activities.

1.3.3 Users

Within this field of application, principal users of this International Standard comprise those involved in processes throughout the enterprise, system and software life cycles, including:

- architects that apply it to assist them to create, express, communicate and document architectures;
- clients, acquirers, users and operators of systems that use it to assist them to understand architectures;
- enterprise stakeholders (such as architects, executives, managers, management staff, suppliers and acquirers) that use it to assist them to understand, interpret, and analyse architecture descriptions to establish, maintain, and transform enterprises;
- system stakeholders (such as architects, designers, programmers, maintainers, testers, domain engineers, quality assurance staff, configuration management staff, suppliers and project managers) that use it to assist them to understand, interpret, and analyse architecture descriptions to develop, deliver, and maintain systems;
- chief information officers, chief engineers, auditors, independent assessors and those who oversee and evaluate systems and their development;
- managers of architecting endeavours who use it to assist in the establishing, planning, monitoring and controlling such undertakings; and
- other people who are involved in enterprise-wide activities that span multiple system developments, including those that seek to establish and codify architecture frameworks, architecture viewpoints and architecting methods meeting the requirements herein.

Secondary users include researchers who can use the standard to provide a common vocabulary for expressing their research discoveries related to novel methods or techniques that enable or improve architecting, architecture governance, management, and enablement practices.

1.3.4 Limitations

This International Standard does not prescribe a specific architecture or system life cycle model, development methodology, method, model or technique. The users of this International Standard are responsible for selecting a life cycle model for the organization or project and mapping the processes, activities, and tasks in this International Standard into that model. The parties are also responsible for selecting and applying appropriate methodologies, methods, models and techniques suitable for the organization or project.

Although this International Standard 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 20000-1 (also published as IEEE Std 20000-1), and the information security management system provided by ISO/IEC 27000.

This International Standard 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 Conformance

NOTE: This text is an adaptation of the conformance clause of the ISO/IEC/IEEE 15288.

The requirements in this International Standard are contained in Clauses 6, 7, 8 and in Annex A and B. It is recognized that particular projects or organizations may not need to use all of the processes provided by this International Standard. Therefore, implementation of this International Standard typically involves selecting and declaring a set of processes, activities, work products and resources suitable to the organization or project. There are two ways that an implementation can be claimed to conform to the provisions of this International Standard – full conformance and tailored conformance.

2.1 Intended usage

his International Standard provides requirements for a number of processes suitable for usage during the life cycle of an enterprise, a project, a system or product. It is recognized that particular projects or organizations may not need to use all of the processes provided by this International Standard. Therefore, implementation of this International Standard 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 International Standard — full conformance and tailored conformance.

There are three criteria for claiming full conformance. Achieving either criterion suffices for conformance, although the chosen criterion (or criteria) is to 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 may be useful for implementing processes to be used in the context of an innovative life cycle model.

- And alternatively, claiming “full conformance to work products” asserts that all of the required work products declared for the overall set of processes and per process are achieved.

NOTE 1 Options for conformance are provided for needed flexibility in the application of this International Standard. Each process has a set of objectives (phrased as “outcomes”), a set of work products, 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, might have innovative process variants that achieve the objectives of the declared set of processes without implementing all of the activities and tasks and delivering all the work products. These users can assert full conformance to the outcomes of the declared set of processes. The three criteria—conformance to task conformance to outcome, and conformance to the work products—are necessarily not equivalent since specific performance of activities and tasks may require, in some cases, a higher level of capability than just the achievement of outcomes and delivery of the work products.

NOTE 3 When this International Standard is used to help develop an agreement between an acquirer and a supplier, clauses of this International Standard can be selected for incorporation in the agreement with or without modification. In this case, it is more appropriate for the acquirer and supplier to claim compliance with the agreement than conformance with this International Standard.

NOTE 4 An organization (for example, national, industrial association, company) imposing this International Standard as a condition of trade can specify and make public the minimum set of required processes, outcomes, work products, activities, and tasks, which constitute suppliers‘ compliance with the conditions of trade.

NOTE 5 Requirements of this International Standard 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.

2.2 Full conformance

2.2.1 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 International Standard 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 ISO/IEC 15504-2 and ISO/IEC 33002. Those standards provide for the assessment of the processes of this International Standard, providing a basis for improvement. Users intending process assessment and improvement may use the process outcomes written in this International Standard as the "process reference model" required by ISO/IEC 15504-2 and ISO/IEC 33002.

2.2.2 Full conformance to work products

A claim of full conformance declares the set of processes for which conformance is claimed. Full conformance to work products is achieved by demonstrating that all of the work products declared for the overall set of processes and per process 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.

2.2.3 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 may be appropriate in contractual situations where an acquirer or a regulator requires detailed understanding of the suppliers' processes.

2.3 Tailored conformance

When this International Standard is used as a basis for establishing a set of processes that do not qualify for full conformance, the clauses of this International Standard are selected or modified in accordance with the tailoring process prescribed in Annex A. The tailored text, for which tailored conformance is claimed, is declared. Tailored conformance is achieved by demonstrating that the outcomes, work products, activities, and tasks, as tailored, have been achieved.

3 Normative references

The following International Standards, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

- ISO/IEC/IEEE 12207:2008 Systems and software engineering — Software life cycle processes
- ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes
- ISO 15704:2000 Industrial automation systems — Requirements for enterprise-reference architectures and methodologies
- ISO/IEC 38500:2008 Corporate governance of information technology
- ISO/IEC/IEEE 42010:2011 Systems and software engineering — Architecture description
- ISO/IEC 42030:201x Systems and software engineering — Architecture evaluation

4 Terms and definitions

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

4-1
activity
1. set of cohesive tasks of a process.
   [SOURCE: ISO/IEC 15288:2015]
2. a component of work performed during the course of a project.
   [SOURCE: A Guide to the Project Management Body of Knowledge (PMBOK® Guide) — Fourth Edition].

4-2
architecting
process of conceiving, defining, expressing, documenting, communicating, certifying proper implementation of, maintaining and improving an architecture throughout the life cycle for the entity being architected.

Note to entry: Architecting takes place in the context of an organization (“person or a group of people and facilities with an arrangement of responsibilities, authorities and relationships”) and/or a project (“endeavour with defined start and finish criteria undertaken to create a product or service in accordance with specified resources and requirements”)

[SOURCE: ISO/IEC/IEEE 12207:2008, ISO/IEC/IEEE 15288:2015, replaced “system” with “entity being architected” to generalize this for items other than systems].

4-3
architecture
fundamental concepts or properties of one or several entities in their intended environment embodied in their elements, relationships, and in the principles governing their design and evolution along their life-cycle.

[SOURCE: ISO/IEC/IEEE 42010:2011, modified to replace “a system” by “one or several entities”]

Note 1 to entry: The entities addressed by architecture can be, for example, a set of capabilities, a complete solution including a system of interest and enabling items, a single system, a product, a service, or a software item.

Note 2 to entry: The entity is a single system when considering system architecture. In that case the [ISO 42010 - 2011] applies: “fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution”.

Note 3 to entry: Term “enterprise Architecture” applies when the architecture is provided with enterprise vision. I.e. the architecture is normally expected to address the whole set of enterprise stakeholders’ concerns. However, it can also focus on a particular domain of interest or concern of the enterprise. For example, this focus can be on the information system, as defined in [TOGAF V9.1]: “structures and gives context to all enterprise activities delivering concrete business outcomes primarily but not exclusively in the IT domain.”

[SOURCE: This standard]

4-4
architecture description
work product used to express an architecture.

[SOURCE: ISO/IEC/IEEE 42010:2011]

4-5
architecture framework
conventions, principles and practices for the description of architectures established within a specific domain of application and/or community of stakeholders

EXAMPLE 1 Generalised Enterprise Reference Architecture and Methodologies (GERAM) [ISO 15704] is an architecture framework.

EXAMPLE 2 Reference Model of Open Distributed Processing (RM-ODP) [ISO/IEC 10746] is an architecture framework.

[SOURCE: ISO/IEC/IEEE 42010:2011]

4-6
architecture view
work product expressing the architecture from the perspective of specific concerns.

[SOURCE: ISO/IEC/IEEE 42010:2011, modified to replace “system concerns” by “concerns”]

4-7
architecture viewpoint
work product establishing the conventions for the construction, interpretation and use of architecture views to frame specific concerns

[SOURCE: ISO/IEC/IEEE 42010:2011, modified to replace “system concerns” by “concerns”]

4-8
concept of operations
verbal and/or graphic statement, in broad outline, of an organization’s assumptions or intent in regard to an operation or series of operations.

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. The concept is designed to give an overall picture of the organization operations.

Note 2 to entry: It provides the basis for bounding the operating space, system capabilities, interfaces and operating environment.

[SOURCE: ANSI/AIAA G-043A-2012e]

4-9
concern
1. (architecture) interest in an architecture relevant to one or more of its stakeholders
   [SOURCE: This standard]
2. (system) interest in a system relevant to one or more of its stakeholders

[SOURCE: ISO/IEC/IEEE 42010:2011]

4-10
customer
organization or person that receives a product or service

EXAMPLE Consumer, client, user, acquirer, buyer, or purchaser

Note 1 to entry: A customer can be internal or external to the organization.
[SOURCE: ISO 9000:2005, modified – added ‘service’].

4-11
enabling system
system that supports a system-of-interest during its life cycle stages but does not necessarily contribute directly to its function during operation

Note 1 to entry: For example, when a system-of-interest enters the production stage, a production-enabling system is required.

Note 2 to entry: Each enabling system has a life cycle of its own. This International Standard is applicable to each enabling system when, in its own right, it is treated as a system-of-interest

[SOURCE: ISO/IEC/IEEE 15288:2015]

4-12
enterprise (as context)
human venture that has definite mission, goals, and objectives to offer products or services, or to achieve a desired mission outcome or business outcome.

Note to entry: In this International Standard, 'enterprise' refers to concrete (e.g. extended supply chain enterprise) or abstract (e.g. virtual enterprise) entities

[SOURCE: ISO 15704 planned evolution]

4-13
enterprise architecture
See Architecture

4-14
life cycle

1. set of distinguishable phases and stages within phases that an entity goes through from its creation until it ceases to exist.
   [SOURCE: ISO 15704 planned evolution]

   NOTE The architecture life cycle starts with the identification of a need for the architecture and ends with its decommissioning/discarding.
   [SOURCE: This standard]

2. evolution of a system, product, service, project or other human-made entity from conception through retirement
   [SOURCE: ISO/IEC/IEEE 15288:2015]

4-15
life history
actual sequence of phases and stages that an entity has gone through during its lifetime.

[SOURCE: ISO 15704 planned evolution]

4-16
model
1. a representation of a real world process, device, or concept.
   [SOURCE: IEEE 1233:1998];

2. abstract description of reality in any form (including mathematical, physical, symbolic, graphical, or descriptive) that presents a certain aspect of that reality.
   [SOURCE: ISO/CEN 19439/40]

   Note to entry: A model is expressed using a standard or de facto modelling language that employs some degree of formalism and a structured or semi-structured grammar. A verbal description, for example, is not necessarily a model.
   [SOURCE; This standard]

4-17
organization
group of people and facilities with an arrangement of responsibilities, authorities and relationships

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

Note 1 to entry: An identified part of an organization (even as small as a single individual) or an identified group of organizations can be regarded as an organization if it has explicitly stated responsibilities, authorities and relationships. A body of persons organized for some specific purpose, such as a club, union, corporation, or society, can be an organization.

[SOURCE: ISO 9000:2005, modified – Note 1 to entry has been added]

4-18
phase
distinguishable part of a sequence or a cycle occurring over time.

[SOURCE: This standard]

4-19
problem
difficulty, uncertainty, or otherwise realized and undesirable event, set of events, condition, or situation that requires investigation and corrective action

[SOURCE: ISO/IEC/IEEE 15288:2015]

4-20
process
set of interrelated or interacting activities that transforms inputs into outputs

[SOURCE: ISO/IEC/IEEE 15288:2015]

4-21
process outcome
observable result of the successful achievement of the process purpose

[SOURCE: ISO/IEC/IEEE 12207:2008]

4-22
process purpose
high level objective of performing the process and the likely outcomes of effective implementation of the process

Note 1 to entry: The implementation of the process should provide tangible benefits to the stakeholders.

[SOURCE: ISO/IEC/IEEE 12207:2008]

4-23
product
1. result of a process.
   [ISO 9000:2005]
2. an artefact that is produced, is quantifiable, and can be either an end item in itself or a component item.
   [SOURCE: A Guide to the Project Management Body of Knowledge (PMBOK® Guide) — Fourth Edition.]
   Note 1 to entry: There are four agreed generic product categories: hardware (e.g., engine mechanical part); software (e.g., computer program); services (e.g., transport); and processed materials (e.g., lubricant). Hardware and processed materials are generally tangible products, while software or services are generally intangible. Most products comprise elements belonging to different generic product categories. Whether the product is then called hardware, processed material, software, or service depends on the dominant element.
   [SOURCE: ISO 9000:2005]
   Note 2 to entry: Goods are tangible product.

[SOURCE: This Standard]

4-24
project

1. endeavour with defined start and finish dates undertaken to create a product or service in accordance with specified resources and requirements.
   [SOURCE: ISO/IEC/IEEE 15288:2015]
2. set of interrelated or interacting activities which transforms inputs into outputs
   [SOURCE: ISO 9000:2005]
3. set of activities for developing a new product or enhancing an existing product.
   [SOURCE: ISO/IEC 26514:2008]
   Note to entry: A project, when it involves a set of architecture activities, will act on incoming architecture work products (when applicable) and generate outgoing architecture work products.
   [SOURCE: This standard]

4-25
quality attribute
feature or characteristic that affects an item's quality.
[SOURCE: ISO/IEC/IEEE 24765:2010 3.2384 2nd alternative.]

4-26
resource
asset that is utilized or consumed during the execution of a process

Note 1 to entry: 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.
[SOURCE: ISO/IEC/IEEE 15288:2015]

4-27
service
means of delivering value for the customer by facilitating results the customer wants to achieve.

Note 1 to entry: Service is generally intangible.

Note 2 to entry: A service can also be delivered to the service provider by a supplier, an internal group or a customer acting as a supplier.
[SOURCE: ISO/IEC 20000-1:2011]

4-28
stage
period within the life cycle of an entity that relates to the state of its description or realization

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

Note 2 to entry: Stages often overlap.
[SOURCE: ISO/IEC/IEEE 15288:2015]

4-29
stakeholder
(architecture) individual, team, organization, or classes thereof, having an interest in an architecture

NOTE As noted in ISO/IEC/IEEE 42030, not all stakeholders for a system are necessarily stakeholders of the architecture related to that system. Likewise, not all stakeholders of an architecture are necessarily stakeholders of systems related to that architecture.
[SOURCE: This Standard]

(system) individual, team, organization, or classes thereof, having an interest in a system
[SOURCE: ISO/IEC/IEEE 42010 – 2011]

4-30
system
Combination of interacting elements organized to achieve one or more stated purposes
[SOURCE: ISO/IEC/IEEE 15288:2015]

4-31
system architecture
See Architecture

4-32
system-of-systems
A large system that delivers unique capabilities, formed by integrating independently useful systems
[SOURCE: ISO 24765:2011]

Note to entry: A system of systems (SoS) brings together a set of systems for a task that none of the systems can accomplish on its own. Each constituent system keeps its own management, goals, and resources while coordinating within the SoS and adapting to meet SoS goals.
[SOURCE: ISO/IEC/IEEE 15288:2015].

4-33
task
required, recommended, or permissible action, intended to contribute to the achievement of one or more outcomes of a process
[SOURCE: ISO/IEC/IEEE 15288:2015]

4-34
trade-off
decision-making actions that select from various requirements and alternative solutions on the basis of net benefit to the stakeholders
[SOURCE: ISO/IEC/IEEE 15288:2015]

4-35
work product
an artifact associated with the execution of a process
[SOURCE: ISO/IEC 15504-1:2004]

5 Key concepts and application of this International Standard

5.1 Introduction

This clause introduces the concepts regarding architecture processes. The concepts introduced in this clause are used in Clauses 6 through 8 to express requirements.

5.2 Contexts of architecture activities

An enterprise is a venture involving a set of stakeholders sharing the same enterprise objectives (goals and missions). Some of these stakeholders have expectations, needs or requirements which scope projects of architecture activities.

These architecture activities are performed according to defined organized processes defined in project and enterprise contexts.

This International Standard provides the requirements for these activities, their outcomes and work products. Conformance and tailoring, as specified in Clause 2 and Annex A, has to be considered in the Architecture Governance and Architecture Management processes.

5.3 Architecture and life concepts

As long as an architecture has to be in line with the entity or entities it describes, a roadmap can be prepared to explain the life history of the architecture. If prepared, the roadmap should describe alternatives of life stories (or scenarios).

With regards to description of an architecture roadmap, a life history is normally a set of epochs which date the architecture phases of the architecture life cycles.

The notion of life cycle stage used in this International Standard is referring to the ISO/IEC/IEEE 15288 life cycle model where stages are related to the critical quality characteristics during the evolution of one or several entities and of their environment (e.g. utilization, support, etc.). Life cycle stages may occur several times during the life cycle phases of the considered entities.

5.4 Object of the architecture

Architecture can be associated different purposes or circumstances, with at least two cases:

- Architecture provides the orientation for a solution along its life cycle. This solution can be one of several systems; or the only definition of the capabilities or services to be provided by this or these systems.

- Architecture emerging from an existing solution. In that case, the architecture corresponds to the perception of the existing system or systems.

NOTE In these two cases, the architecture and the solution are maintained consistently: with a top-down approach the architecture drive the system(s) life cycle; and with a bottom-up approach the system(s) leverage the stakeholders' vision formalized in the architecture.

5.5 Architecture kinds

Architectures can be:

- Prescriptive: for definition of requirements or at least expectation regarding one or several entities and their environments.

- Contractual: for committed statements regarding one or several entities and their environments.

- Descriptive: for definition of a solution corresponding to an expectation.

- Compulsory: for definition of reference data to be used by either activities or entities.
- Illustrative: for description of existing or already defined entities.

Annex D gives more information regarding different architecture kinds covering these different cases It also provides information regarding architecture styles and motivation models.

5.6 Architecture and architecture description

The conceptual model for an architecture description [ISO/IEC/IEEE 42010 Architecture and architecture description] applies to this standard. However, it can be the case that an architecture can be started or completed before any system identification has occurred. In that case, the architecture will context for the system(s) to be identified later.

Note that in studying an existing system there is a clear distinction between producing an architecture description document and comprehending the pre-existing architecture of the system. One may have a clear comprehension of the architecture of the system of interest without having developed a complete document. Conversely, one may have a comprehensive document (albeit one badly written) and still have no clear comprehension of any organizing structure or principles of the system (perhaps because there are no such organizing abstractions).

5.7 Process concepts

NOTE This section is adapted from ISO/IEC/IEEE 15288.

5.7.1 Criteria for processes

The determination of architecture processes in this International Standard is based upon three basic principles:

- Each architecture process has strong relationships among its outcomes, activities and task.
- 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.

5.7.2 Description of processes

Each process of this standard 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, recommendations, or permissible actions intended to support the achievement of the outcomes.

Additional detail regarding this form of process description can be found in ISO/IEC TR 24774.

5.7.3 General characteristics of processes

In addition to the basic attributes described in the previous sub-clause, processes may be characterized by other attributes common to all processes. ISO/IEC 15504-2 identifies common process attributes that characterize six levels of achievement within a measurement framework for process capability. Annex C in ISO/IEC/IEEE 15288 includes the list of process attributes that contribute to the achievement of higher levels of process capability as defined in ISO/IEC 15504-2

5.7.4 Tailoring

Annex A, which is normative, defines the basic activities needed to perform tailoring of this International Standard. Note that tailoring may diminish the perceived value of a claim of conformance to this standard. This is because it is difficult for other organizations to understand the extent to which tailoring may have removed desirable provisions. An organization asserting a single-party claim of conformance to this standard may find it advantageous to claim full conformance to a smaller list of processes rather than tailored conformance to a larger list of processes.

5.8 Process application

The architecture processes defined in this International Standard can be used by any organization when acquiring, using, creating, or supplying a system, as well as when operating, evolving, or transforming an enterprise. They can be applied at any level in an enterprise and at any stage in the life cycle of the architecture or associated systems.

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 architecture process in **Figure 1** can be invoked, as required, at any time throughout the life cycle. The order that the processes are presented in this standard does not imply any prescriptive order in their use. However, sequential relationships are introduced by the definition of a life cycle model. The detailed purpose and timing of use of these processes throughout the life cycle are influenced by multiple factors, including social, trading, organizational and technical considerations, each of which can vary during the life of an architecture. An individual architecture life cycle is thus a complex aggregation of processes that will normally possess concurrent, iterative, recursive and time dependent characteristics.

Concurrent use of processes can exist within an enterprise or a project (e.g., when the current architecture is being implemented at the same time that a future architecture is being formulated for the same system), and between projects (e.g., when architecture entities are developed at the same time under different project responsibility).

When the application of the same process or set of processes is repeated on the same architecture, the application is referred to as iterative. The iterative use of processes is important for the progressive refinement of process outputs, e.g., the interaction between successive architecture evaluation efforts can incrementally build confidence in the suitability of the architecture. 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 architecture objectives, stakeholder needs, analysed risks or opportunities. Such questions should be resolved before completing the activities of a process or set of processes.

The recursive use of processes, i.e., the repeated application of the same process or set of processes applied to successive levels of architecture entities in an architecture's structure, is a key aspect of the application of this International Standard. The outputs of processes at any level, whether information, artefacts or services, are inputs to the processes used at the level below (e.g., during system realization) or level above (e.g., during enterprise transformation). The outcomes from one application are used as inputs to the next lower (or higher) architecture in the architecture structure to arrive at a more detailed or mature set of outcomes. Such an approach adds value to successive architectures in the architecture structure.

The changing nature of the influences on the architecture (e.g., operational environment changes, new opportunities for architecture entity 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 architecture. 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 architecture 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 and/or the entry criteria of the formal progress reviews within that stage.

The discussion in this section on iterative and recursive use of the architecture processes is not meant to imply any specific hierarchical, vertical, or horizontal structure for the architecture, system-of-interest, enabling system, organization, or project.

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

Further elaboration of these concepts can be found in the ISO/IEC/IEEE TR 24748 guides, on the application of architecture processes.

5.9 Process reference model

Annex C in ISO/IEC/IEEE 15288 defines a Process Reference Model (PRM) at a level of abstraction higher than that of the detailed requirements contained in the main text of this International Standard. The PRM is applicable to an organization that is assessing its processes in order 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 evaluation.

5.10 Processes in this standard

Each of the architecture processes in this International Standard is described in terms of its purpose and desired outcomes and list activities and tasks that need to be performed to achieve those outcomes. The architecture processes are depicted in **Figure 1**. The processes described in this International Standard are not intended to preclude or discourage the use of additional processes that organizations find useful.

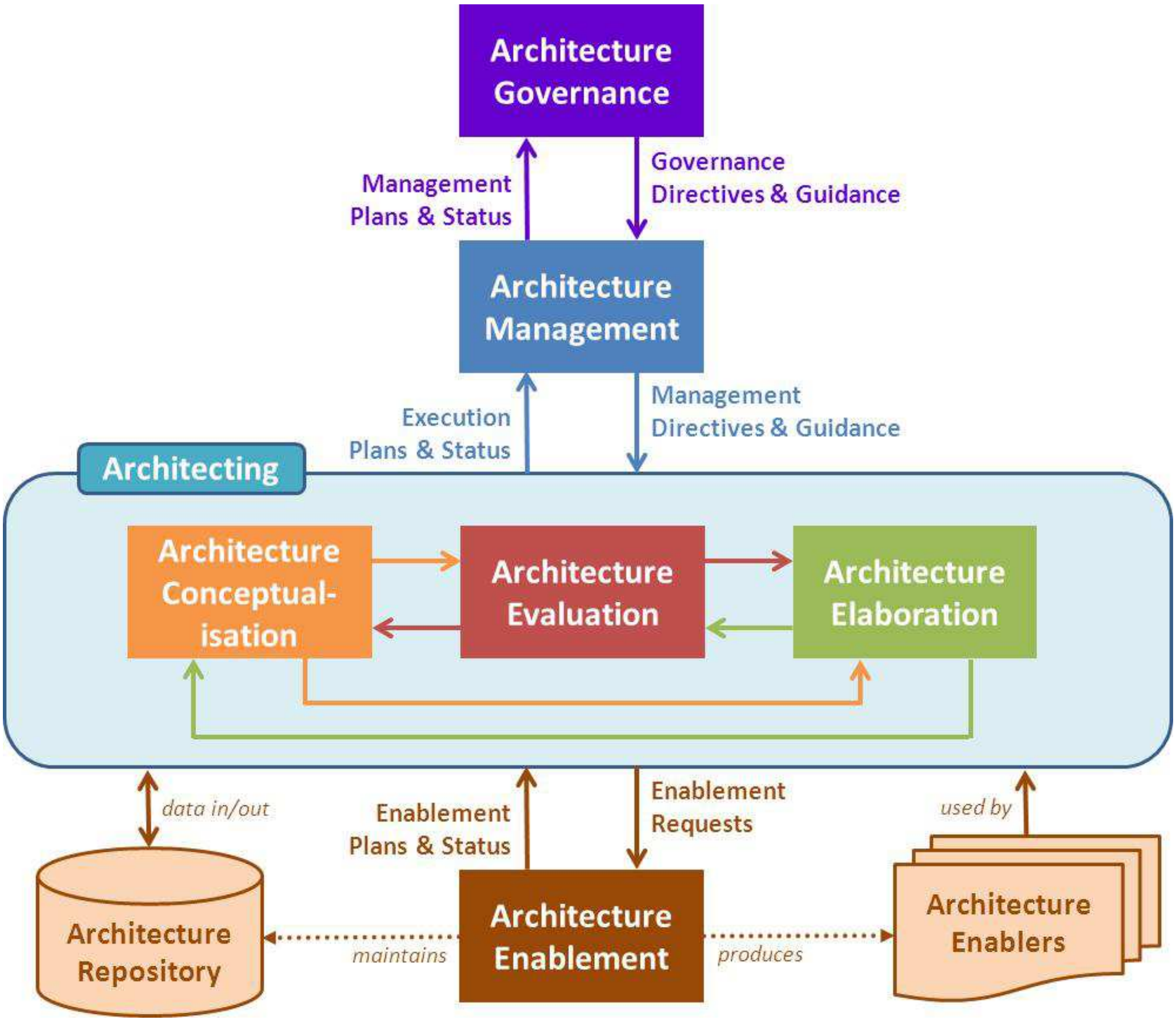


Figure 1: Architecture process interactions

The key information flows between the architecting processes are illustrated in **Figure 2**. This illustration goes beyond the detail specified in the process requirements in clause 6 since it serves the purpose of providing a sense of how these processes work together. Note that this does not show the flow of work products but rather the key information related to these work products.

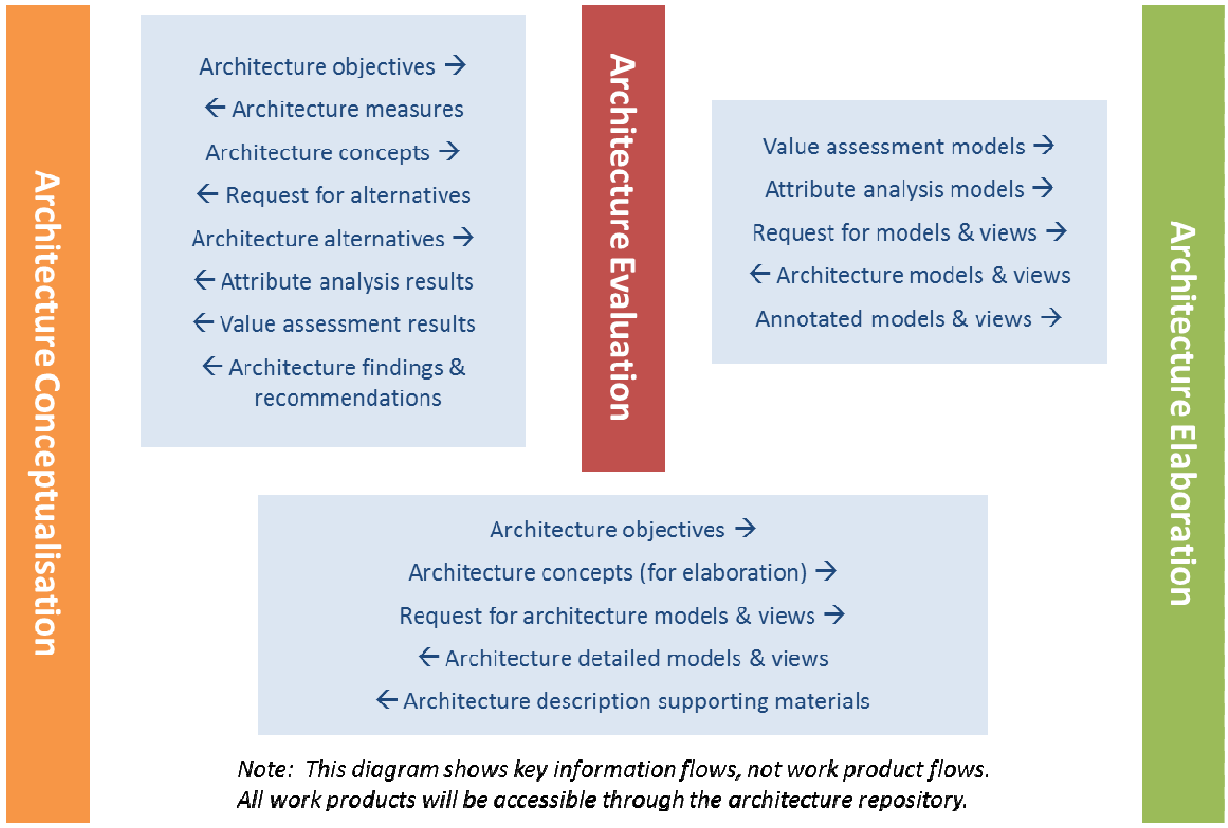


Figure 2: Key information flows between the architecting processes

6 Processes

This clause specifies the requirements for the architecture processes listed in **Table 1**.

Table 1: Architecture processes

Name	Purpose
Architecture Governance	Establish standards and policies related to one or more architectures of interest and their development, and to monitor and facilitate the alignment of the architecture(s) to stakeholder concerns, policies and standards, including organizational and environmental constraints.
Architecture Management	Ensure execution of directives for development of the architectures, to ensure that the development runs according to these directives, to the expected timetables, to the assigned budgets, and that the architecture satisfies its objectives.
Architecture Conceptualisation	Generate architecture alternatives, to select one or more alternatives that address stakeholder concerns and meet relevant requirements, and to express them in a set of consistent views.
Architecture Evaluation	Determine the degree to which the architecture meets architecture objectives and addresses stakeholder concerns
Architecture Elaboration	Create one or more architecture descriptions in a form that uses established notations and languages and captures this in a set of consistent views and models.
Architecture Enablement	Ensure that the supporting capabilities needed to perform the Architecture activities in the other architecture processes are available when and where necessary.

These processes can be performed at any level within the enterprise.

These processes interact as illustrated in Figure 1.

NOTE 1 The architecture processes are performed concurrently even if the governance and management directions circulate in down-flows and operation reports in up-flows.

NOTE 2 The architecture processes can be performed concurrently, with interactions between them and iteration happening over time.

NOTE 3 The Architecture Enablement process sustains the architecture processes with a set of coherent architecture enablers in order to ensure a seamless and consistent set of services and data in the architecting environment.

6.1 Architecture governance process

6.1.1 Purpose

The purpose of the Architecture Governance process is to establish standards and policies related to one or more architectures of interest and their development, and to monitor and facilitate the alignment of the architecture(s) to stakeholder concerns, policies and standards, including organizational and environmental constraints.

Governance is needed for consistent management, cohesive standards and policies, proper guidance, uniform processes and appropriate decision-rights. It is defined as a given area of responsibility and for proper oversight and accountability. This allows to identify, manage, audit, and disseminate all information related to architecture decisions, management actions in response to these decisions, contracts affecting the architecture(s), and implementation of architecture changes.

An overarching architecture governance model is needed for the establishment of an integrated composition of architectures under development by teams involved in projects that impact one or more systems.

Architecture governance has oversight over the architecture objectives for the architectures of interest to ensure their consistency.

Each set of architecture objectives will be considered with respect to factors of maintenance, servicing, and upgrade with minimal disruption of the everyday operations. In particular, consideration will be given to available internal and external resources in order to determine when general resources can be adapted for specific needs and to determine where specific solutions can be generalized to support wider re-use.

An architecture governance framework defines conceptual and organizational structures, and the relationship to the project management governance framework. It provides a structured, decision-making approach.

Architecture governance monitors the portfolio of architectures to ensure that the approved approach is being applied correctly across all phases of architecture development efforts.

NOTE 1 Architecture governance is typically performed at higher levels of the organization providing oversight over concurrent business units, programs, projects, etc. Architecture governance has responsibilities for legal compliance, alignment with organizational goals and objectives, optimum utilization of resources, maintaining focus on the long term vision, responding to changes in the marketplace and user community, anticipating new forces and scenarios that will likely arise, maximizing shareholder gains, etc.

NOTE 2 Architecture governance acts on a portfolio of architectures in order to check the alignment between them and for compliance with the organizational mandates and expectations. Usually the portfolio consists of several architectures that are related to each other, but the portfolio could consist of a single architecture if appropriate.

NOTE 3 To be consistent with ISO 38500 for IT governance, the activities of architecture governance include the following kinds of activity:

a) Evaluate coherency of the roadmap of the architecture(s) toward desired outcomes
b) Direct changes to the architecture(s)
c) Monitor architecture collection to ensure compliance with the governance directions

NOTE 4 Each activity is governed by principles. An organizational authority should be in charge of checking that the activities are performed according to these principles. This authority could be called "Design Authority", identified for governance according to architecture principles with an escalation approach when necessary.

6.1.2 Outcomes

An implementation of the Architecture Governance process shall achieve the following outcomes:

a) Current business needs are fulfilled.

b) Provisions are made to address anticipated future business needs.

c) Business resources are balanced to best meet business objectives.

6.1.3 Activities and tasks

The organization or project shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the Architecture Governance process.

6.1.3.1 Prepare for architecture governance

a) Identify architectures of interest that require governance oversight.
b) Define the governance roles and authorities with respect to the architectures of interest.
c) Establish responsibilities, accountabilities, structures and behavior to support the Architecture Governance process and reporting requirements.
d) Establish guiding principles and instructions for performing architecture governance.
   NOTE The instructions need to specify the extent to which the activities and tasks defined in this IS for architecture governance are to be followed for the enterprise.
e) Establish the architecture governance organization that is consistent with the defined roles, authorities, responsibilities and accountabilities.
f) Establish metrics for the architecture governance effort.
g) Establish decision forums to carry out architecture governance instructions.
h) Define uniform procedures for identifying, managing, auditing and disseminating information related to architecture decisions.
   i) Link these procedures to architecture strategies, policies and objectives.
   ii) Map governance procedures to resources and constraints to support strategy, planning and decision making.
i) Plan the architecture governance effort.

6.1.3.2 Establish desired strategic outcomes

a) Examine current and future business needs.
   i) Examine current and future organizational objectives that must be achieved, such as maintaining competitive advantage.
   ii) Examine and make judgment on the current and future use of systems, including strategies, proposals and supply arrangements (whether internal, external, or both) in support of the

organizational objectives.

NOTE The examination should consider the external or internal pressures acting upon the enterprise, such as technological change, economic and social trends, and political influences.

iii) Examine specific objectives of the strategies and proposal they are evaluating.

b) Examine current and future mission needs for those mission supported by the enterprise.
   i) Examine current and future mission objectives that must be achieved, such as improving the timely delivery of products.
   ii) Examine and make judgment on the current and future use of systems, including strategies, proposals and supply arrangements (whether internal, external, or both) in support of the mission objectives.

   NOTE The examination should consider the external or internal pressures acting upon the enterprise, such as technological change, economic and social trends, and political influences.

   iii) Examine specific objectives of the strategies and proposal they are evaluating.
c) Set the vision that creates decision making power and accountability.
   i) Oversee a governance model that enables integrated composition of architectures of interest.
   ii) Implement a governance framework that supports this governance model so as to define conceptual and organizational structures and provide a structured, decision making approach.
   iii) Establish a decision making mechanism that minimizes or avoids potential conflicts of interests with escalation in the organization if the problems cannot be properly addressed by architecture governance.
d) Set the architecture strategy; make the relevant decisions for the architectures of interest.
   i) Discover, develop, define and evaluate the goals of the architectures of interest.
   ii) Define architecture governance policies related to one or more architectures of interest.
   iii) Define architecture objectives for the architectures of interest that support the architecture goals.
   iv) Define a comprehensive set of architecture principles that lead the governance activities.
   v) Determine adherence of objectives based on governance compliance criteria and strategies.
   vi) Establish management criteria for control of architecture governance practices, dispensations, and compliance.
e) Identify the architecture goals and objectives to be pursued and desired levels of achievement for each.
   i) Identify work to be performed to achieve these goals and objectives.
   ii) Create a configuration of resources which are necessary ingredients in meeting these goals and objectives.
   iii) Establish means of measuring governance efficiency.
   iv) Establish a governance dashboard for monitoring compliance with governance directives and guidance.

6.1.3.3 Direct changes to relevant architectures

a) Assign responsibility for, and direct preparation and implementation of, policies that set the direction for investments in system development projects and system operations activities.
b) Ensure that the transition of systems to operational status is properly planned and managed, according to business transformation within the organization, taking into account impacts on business and operational practices as well as existing systems and infrastructure.
c) Encourage a culture of good governance of architectures in their organization by requiring managers to provide timely information, to comply with direction and to conform with the established principles of governance.
d) Direct the submission of proposals for approval to address identified needs.
e) Review against the recommendation.
f) Make strategic decisions within scope of architecture generation responsibilities and authorities.
g) Elevate other decisions to the appropriate organizational decision forum.
h) Communicate the decisions.

6.1.3.4 Monitor and assess compliance with governance direction

a) Monitor through appropriate measurement means, the performance of the systems developed or operated by the enterprise.

NOTE The architectures are not themselves monitored. In order to see how well the architectures are helping to achieve the enterprise goals and objectives, architecture governance must monitor metrics from the actual systems that derive from the architecture.

b) Ensure that measured performance is in accordance with plans, particularly with regard to meeting business objectives.

NOTE Performance analysis includes analysis of the resources put in place.

c) Ensure that the architectures and systems conform to external obligations (regulatory, legislative, common law, contractual) and internal work practices.

NOTE 1 Responsibility for specific aspects of the architectures and systems can be delegated to managers within the organization. However, accountability for the effective, efficient and acceptable use and delivery of system capabilities remains with the directors and cannot be delegated.

NOTE 2 Systems themselves are sometimes directly assessed to see if they are conforming to the obligations (which are often spelled out in the architecture views). It is often not adequate to merely check to see if the architecture is compliant because sometimes the architectural features don't get properly translated into system design or operations.

d) Conduct a compliance assessment for the architectures of interest.
e) Define checks and balances that enable effective monitoring of management of the architectures of interest.
f) Perform dependability analysis to exploit the associations amongst the architectures of interest.
g) Ensure that the governance policies are adopted.
h) Monitor and assess metrics for the architecture governance effort.
i) Scrutinize the effectiveness of the governance policies for one or more architectures of interest and initiate steps to revise the policies as necessary.
j) Collect lessons learned regarding architecture governance as reported from the other architecture processes and make these lessons learned available to future projects.

6.1.4 Work products

Architecture governance plans shall be compliant with ISO ???.

Architecture governance plans shall be compliant with enterprise and project policies and rules.

Architecture governance plans shall be stored in the organization-designated architecture repository.

Architecture governance directives and guidance shall be compliant with ISO ???.

Architecture governance directives and guidance shall be stored in the organization-designated architecture repository.

Management plans and status provided to architecture governance shall be stored in the organization-designated architecture repository.

6.2 Architecture management process

6.2.1 Purpose

The purpose of the Architecture Management process is to ensure execution of directives for development of the architectures, to ensure that the development runs according to these directives, to the expected timetables, to the assigned budgets, and that the architecture satisfies its objectives.

This process also includes necessary changes to the planned activities for developing the architecture to adapt to drift and variations in related activities. The modifications of the plan can sometimes lead to a revision of the architecture.

Execution of the Architecture Management process provides the following benefits:

- Description of all the system elements that satisfy the requirements of the system are present.
- A foundation for assessing the elements of the system is established.
- The evaluation of architecture alternatives towards selection of the best compromise is performed.
- Corrective actions are defined and executed when the architecture does not satisfy the planned objectives.
- The architecture is reformulated when the objectives or the constraints associated with the architecture are changed or when the working hypotheses are found to be erroneous.
- The promotion of the development of the architecture is authorized when its benefits and risks are sufficiently defined per milestone to provide confidence.
- The technical objectives are reached.

NOTE 1 Architecture management is concerned with managing the architecture itself. The main focus of architecture management is on managing the implementation and evolution of the architecture(s) to maximize alignment with strategic goals and objectives.

NOTE 2 The Architecture Management process does not manage the development of the architecture but rather its evolution and its implementation in the design, build, deployment, operations, maintenance, decommissioning, etc. of one or more systems related to the architecture.

NOTE 3 Architecture management is responsible for implementing the guidance and direction from architecture governance where this is accomplished by giving management guidance and direction to the other architecture processes. Architecture management provides status and intentions to architecture governance on how well the architectures are evolving and being implemented.

6.2.2 Outcomes

An implementation of the Architecture Management process shall achieve the following outcomes:

a) Architecting activities are in alignment with architecture governance directives and guidance.

b) Architecture processes are more efficiently and effectively performed.

c) Resulting architectures have greater impact on operations of the enterprise.

6.2.3 Activities and tasks

The organization or project shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the Architecture Management process.

6.2.3.1 Prepare for architecture management

a) Identify architecture management philosophies, tools and techniques according to architecture governance policies, directives and guidance.
b) Identify architectures of interest that require management oversight.
c) Develop an architecture management charter.
   i) Identify stakeholders and their concerns.
   ii) Identify organizational assets (people, resources, processes) that influence architecture management.
   iii) Identify knowledge assets (historical information, issues and defect resolutions, successful outcomes) that can aid in architecture management.

    iv) Identify tools and techniques that can be used for architecture management.
    v) Identify internal and external factors that influence architecture management.
    vi) Decide on a management model (the management philosophy to be utilized).
    vii) Identify the management areas of the architectures of interest and the management criteria.
    viii) Develop a statement of work about the architecture.
    ix) Define measurable architecture management objectives and related success criteria.

d) Develop the architecture management organizational structure.
    i) Identify the possible managerial roles, responsibilities and authorities that are concerned with or involved in architecture management.
    ii) Define an architecture management control hierarchy corresponding to the different roles and responsibilities.
    iii) Ensure proper delegation of management responsibilities in the architecture management control hierarchy.
    iv) Identify degrees of freedom for management role players at all levels of architecture management control hierarchy.
    v) Ensure proper allocation of roles to identified role players in the management hierarchy.

6.2.3.2 Plan for architecture management execution

a) Develop architecture management plans in accordance with governance directions for the architecture of Interest.
    i) Identify architecture requirements and the various concerns.
    ii) Define an architecture management scope statement (scope description, acceptance criteria, architecture deliverables, exclusions, constraints, assumptions).
    iii) Identify specific goals of the architectures that need to be managed and the reasons for their selection.
    iv) Optimize and prioritize the goals in terms of their importance in achieving architecture management objectives.
    v) Create a work breakdown structure that provides a common framework for the overall planning and control.
        (1) Outline the tasks (course of action) required to be performed to achieve these goals, evaluate them and order them in terms of their importance in achieving the goals.
        (2) Define organizational resources necessary for carrying out the tasks.
        (3) Establish evaluation methods for assessing the progress of the tasks towards goal completion.
        (4) Identify alternate courses of action (contingency plan) in case certain goals prove to be unattainable.
        (5) Ensure that the work breakdown structure associated with development of architecture-related systems, when applicable, is consistent with the architecture.
    vi) Define the management process for the architectures of interest based on the identified work breakdown structure.
        (1) Define the processes involved in defining and controlling what is or is not included as part of the architectures of interest.
        (2) Define the actions that are required to ensure that the life cycle processes of architectures of interest are achieved on a timely basis.
        (3) Establish metrics for the architecture management effort.
    vii) Establish the resources necessary for performing the management process and assign responsibility and authority to the management hierarchy.
        (1) Develop the resource breakdown structure that is essential for delivering the outcomes identified in the architecture management plan.
    viii) Define management measures that allow management compliance for the architectures of interest.
        (1) Define measurement systems that can aid in measurement of architectural progress.

b) Develop the architecture management schedule in accordance with the architecture management plan.
    i) Develop a budget to create and manage the architectures of interest and incorporate checks and balances to validate the budget.

   ii) Develop the schedule for the activities identified in the work breakdown structure and define precise and measurable milestones.
   iii) Define the organizing logic and order of the activities and resources necessary for creating and managing the architectures of interest.
   iv) Estimate the duration of each of the activities and include them as part of the schedule.
   v) Optimize the overall management schedule and baseline it.
   vi) Report lessons learned to the architecture governance practices in order to have these lessons available for future programs to apply.

c) Adapt the architecture management plan according to new information.
   i) Adjust management activities, schedule, directions and goals in response to new information.
   ii) Prepare adaptive management actions that respond to new problems or opportunities.
   iii) Apply new knowledge, insights and technologies that contribute to achieving architecture management objectives.
   iv) Ensure that the architectures associated with the enabling systems are included.

6.2.3.3 Implement the architecture management plan

a) Assign resources to all the identified roles in accordance with the sequence of tasks that needs to be performed.
b) Setup the necessary tracking systems which can capture work performance information.
c) Perform the tasks defined in the architecture management plan to achieve the architecture objectives.
d) Capture work performance information and use it to control architecture development.
e) Assess management measures, resource utilization, probable risks, and new opportunities and manage changes to optimize work performance.
f) Make relevant information available to all stakeholders as outlined in the architecture management plan.
g) Continually confirm that work performance results conform to the architecture requirements including deviations and dispensations.
h) Set tactical directions for the architectures of interest that addresses policy decisions.
i) Establish plans for the development or revision of architectures.

6.2.3.4 Monitor and assess compliance with management direction

a) Monitor and control architecture work by tracking, reviewing, and regulating the progress to meet the management objectives as per the architecture management plan.
   i) Monitor work performance to consistently identify variances from architecture management plan.
   ii) Monitor management issues and recommend preventive action in anticipation of possible problems.
   iii) Monitor management plan execution progress against the schedule estimates.
   iv) Monitor health of the architecture (using the identified management measures) and identify any areas that require additional attention.
   v) Monitor the status of the architecture scope and manage changes to the scope baseline.
   vi) Monitor the status of the architecture schedule and manage changes to the schedule baseline.
b) Assess actual outcomes to planned targets and make corrective actions when necessary.
   i) Compare actual performance with planned performance and take corrective action to yield desired performance.
c) Collect and communicate performance information to all relevant stakeholders at periodic intervals.
   i) Maintain accurate timely information concerning the architecture.
   ii) At periodic intervals check to see if the architecture management objectives are being reached.
   iii) Provide forecasts to update current schedule information.
d) Define quality assurance actions and audits that confirm execution of the architecture management plans.

6.2.3.5 Close and prepare for the management plan change

a) Close execution of the architecture management plan.

b) Review all information to assert that the architecture work is complete and that the architecture objectives have been met.
c) Monitor and assess metrics for the architecture management effort.
d) Check compliance with regulations and standards that affect the architectures of interest.
e) Establish procedures to investigate and document the various reasons for management actions taken as part of architecture management.
f) Record lessons learnt and communicate to all relevant stakeholders.
   i) Contribute to best practices for architecture management.
   ii) Scrutinize effectiveness of management philosophies that was adopted in order to address the architecture management problem.
g) Identify changes to be made to the architecture management plan.
h) Select the changes to be made in the next iteration of the architecture management plan.
i) Incorporate the changes into the architecture management plan

6.2.4 Work products

Architecture management plans shall be compliant with ISO ???.

Architecture management plans shall be stored in the organization-designated architecture repository.

Architecture management directives and guidance shall be compliant with ISO ???.

Architecture management directives and guidance shall be stored in the organization-designated architecture repository.

Execution plans and status provided to architecture management shall be stored in the organization-designated architecture repository.

6.3 Architecture conceptualisation process

6.3.1 Purpose

The purpose of the Architecture Conceptualisation process is to generate architecture alternatives, to select one or more alternatives that address stakeholder concerns and meet relevant requirements, and to express them in a set of consistent views.

6.3.2 Outcomes

An implementation of the Architecture Conceptualisation process shall achieve the following outcomes:

a) Key stakeholder concerns have been addressed by the architecture.
b) Expressions of the architecture's key concepts and properties are captured in a form suitable for use by others.
c) Architecture principles and objectives are clearly expressed and understandable by relevant parties.

6.3.3 Activities and tasks

The organization or project shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the Architecture Conceptualisation process.

6.3.3.1 Prepare for architecture conceptualisation

a) Identify the potential problem area(s) that can be addressed by an architecture.
b) Define the expected purpose, scope, objectives, and level of detail of the architecture

conceptualisation effort.

c) Define an architecture conceptualisation strategy that is consistent with the architecture governance and management directives and guidance and is consistent with the purpose, scope and objectives.
d) Select or develop the requisite architecture conceptualisation methods and tools.
e) Plan the architecture conceptualisation effort.
f) Establish metrics for the architecture conceptualisation effort.
g) Collect the data and information needed for the architecture conceptualisation effort.
h) Obtain access to enablers needed for the architecture conceptualisation effort.
   NOTE The enablers will usually be obtained from the Architecture Enablement process. When enablers are obtained from elsewhere, these can become candidate enablers for use by other projects through the Architecture Enablement process.

6.3.3.2 Manage architecture conceptualisation activities

a) Report architecture conceptualisation activity plans and status.
b) Monitor and assess whether architecture governance directives and guidance are being followed.
c) Monitor and assess whether architecture management directives and guidance are being followed.
d) Monitor and assess metrics for the architecture conceptualisation effort.
e) Identify and assess risks and opportunities associated with the architecture conceptualisation effort.
f) Maintain traceability of architecture conceptualisation results to the source material used during the process.
g) Ensure personnel are trained in the use of identified techniques, methods and tools.
h) Ensure that relevant technical, project and organizational processes are properly using architecture conceptualisation products.
   NOTE See section 7.1 for interactions with system life cycle processes.
i) Ensure that relevant enterprise processes are properly using architecture conceptualisation products.
   NOTE See section 7.2 for interactions with enterprise life cycle processes.

6.3.3.3 Analyze problematic situation(s)

a) Identify current and projected problematic situation(s).
b) Identify relevant aspects of the problematic situation(s).
c) Identify strengths, weaknesses, opportunities and threats for the current and projected problematic situations.
d) Identify problems or difficulties in the current and projected problematic situation(s).
e) Identify corresponding stakeholders and their concerns.
f) Understand how the problems or difficulties affect different stakeholders and their priorities in addressing them.
g) Understand complexities of each problem or difficulty, its cause and effect, and how it is being addressed currently.
h) Determine bounding conditions, root causes, and relevant scenarios for each identified problem or difficulty.
i) Identify relevant assumptions, alterables, constraints, conditions and challenges.
j) Develop a problem tree showing the cause and effect relationships for the identified problem(s) or difficulties.
k) Formulate a clear statement of the problem(s).

6.3.3.4 Synthesize potential solution(s)

a) Develop an objective tree by reformulating the elements in the problem tree as conditions and objectives to be achieved.
b) Identify intervention means that can achieve the objectives and serve as potential solution(s).
c) Review the resulting means-ends relationships to assure the completeness of the potential solution(s).
d) Formulate purpose statement(s) for each potential solution.
e) Identify needs, wants and expectations for each potential solution.
f) Identify relevant critical success factors and key performance indicators.

g) Understand stakeholders' value creation context and formulate value propositions for each potential solution.
h) Identify strengths, weaknesses, opportunities and threats for each potential solution.
   i) Identify and characterize risks for each potential solution.
   ii) Identify assumptions with respect to each potential solution.
   iii) Identify additional problems that might be caused by each potential solution.
i) Harmonize elements of each potential solution to ensure that it can be realized in a coherent and cohesive manner.
j) Determine gaps or shortfalls of current or planned solutions in addressing the problem.
k) Formulate a roadmap to address these gaps or shortfalls.

6.3.3.5 Formulate potential architecture(s)

a) Establish and document the desired functional and non-functional characteristics based on the potential solution(s).
b) Devise structural, behavioral and organizational constructs that support the desired functional and non-functional characteristics.
c) Identify key characteristics that provide insight into the architecture and use it to define the context and scope of the architecture.
d) Formulate principles, guidelines, protocols and standards for this architecture.
e) Divide and allocate the characteristics into components and processes that they need to support.
f) Identify the processes and activities that are arranged in a specific order which when activated achieves the identified characteristics.
g) Identify rules governing the components, their composition, interaction, and interdependence that ensures that the architecture provides or enables the desired characteristics.
h) Validate that the architecture provides the desired characteristics or enables them to be realizable.
i) Identify issues and areas for improvement in the architecture(s).

6.3.3.6 Express architecture concepts and properties

a) Define the purpose, scope, breadth and depth for the architecture expression.
b) Specify the form of expression for the architecture suitable for its intended users for this stage of development.
c) Document architecture decisions and architectural characteristics in the specified form.
d) Document key architectural concepts, properties of interest, rationales, conditions, constraints, and assumptions in the specified form.
e) Document architectural guidelines, principles, protocols, and standards in the specified form.
f) Document components, their composition, interdependence, and their interactions in the specified form.
g) Document identified processes, activities, and tasks that aid in characteristics achievement.
h) Develop architecture description consisting of relevant viewpoints, views, models, model correspondences and express them in the specified form.

NOTE This activity could use the results of the Architecture Elaboration process when appropriate. Architecture conceptualisation only needs to describe the architecture to a degree that is suitable for its intended users, which in many cases does not require significant elaboration. The elaboration of architecture views, models, and descriptions will often occur later in the life cycle of the architecture after the architecture has become more mature and the extra effort of elaboration becomes worthwhile. Or elaboration could occur only after several architecture alternatives have been examined for their suitability and they are selected down to one or a few alternatives for further examination and eventual use downstream in the engineering effort.

6.3.3.7 Handoff architecture to downstream users

a) Identify users of architecture information, including relevant architecture descriptions.
b) Prepare architecture description and supporting material for use by others.

c) Deliver architecture information to intended users.
d) Monitor use of architecture information to collect feedback on the architecture and on the form and contents of the architecture work products.

6.3.4 Work products

Architecture descriptions should be compliant with ISO/IEC/IEEE 42010.

Architecture descriptions should be stored in the organization-designated architecture repository.

NOTE During early phases it is sometimes important to be agile and quick in conceptualizing many alternative architectures. Some of these early architecture descriptions will be little more than sketches. After doing several quick rounds of evaluation, then there might be a smaller number of viable architectures that are worthwhile capturing in a more complete form and storing these in the repository for later use.

6.4 Architecture evaluation process

6.4.1 Purpose

The purpose of the Architecture Evaluation process is to determine the degree to which the architecture meets architecture objectives and addresses stakeholder concerns.

NOTE The process activities are based on the concepts and principles specified in the ISO/IEC 42030 standard on architecture evaluation.

EDITORIAL NOTE: The 42030 standard is expected to be released in late 2015.

6.4.2 Outcomes

An implementation of the Architecture Evaluation process shall achieve the following outcomes:

a) Value of the architecture(s) to relevant stakeholders is (are) understood.
b) The basis for evaluation findings and recommendations are clearly communicated and understood by the relevant decision makers and key stakeholders.
c) Relationship between stakeholder concerns and evaluation findings and recommendations is well established.
d) The costs, risks, and tradeoffs associated with implementing the architecture(s) is (are) understood and well founded.

6.4.3 Activities and tasks

The organization or project shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the Architecture Evaluation process.

NOTE This clause gives the requirements on architecture evaluation activities while ISO/IEC 42030 provides the requirements on architecture evaluation elements.

6.4.3.1 Prepare for architecture evaluation

a) Identify the potential decision(s) that can be addressed by the architecture evaluation effort.
b) Define the expected purpose, scope, objectives, and level of detail of the architecture evaluation effort.
c) Define an architecture evaluation strategy that is consistent with the architecture governance and management directives and guidance and is consistent with the purpose, scope and objectives.
d) Select or develop the requisite architecture evaluation methods and tools.

e) Plan the architecture evaluation effort.
f) Establish metrics for the architecture evaluation effort.
g) Collect the data and information needed for the architecture evaluation effort.
h) Obtain access to enablers needed for the architecture evaluation effort.
   NOTE The enablers will usually be obtained from the Architecture Enablement process. When enablers are obtained from elsewhere, these can become candidate enablers for use by other projects through the Architecture Enablement process.
i) Review stated purpose, scope and objectives with the sponsor, architect and other interested parties.
j) Develop a work plan for the evaluation activity in accordance with ISO/IEC 42030.

6.4.3.2 Manage architecture evaluation activities

a) Report architecture evaluation activity plans and status.
b) Monitor and assess whether architecture governance directives and guidance are being followed.
c) Monitor and assess whether architecture management directives and guidance are being followed.
d) Monitor and assess metrics for the architecture evaluation effort.
e) Identify and assess risks and opportunities associated with the architecture evaluation effort.
f) Maintain traceability of architecture evaluation results to the source material used during the process.
g) Ensure personnel are trained in the use of identified techniques, methods and tools.
h) Ensure that relevant technical, project and organizational processes are properly using architecture evaluation products.
   NOTE See section 7.1 for interactions with system life cycle processes.
i) Ensure that relevant enterprise processes are properly using architecture evaluation products.
   NOTE See section 7.2 for interactions with enterprise life cycle processes.

6.4.3.3 Determine evaluation criteria

NOTE 1 The evaluation criteria consist of both the value assessment criteria and the architecture analysis criteria.

NOTE 2 The evaluation objectives might or might not be related to the objectives used in generation of the architecture(s); but in any case the generation objectives need to be examined to determine to what extent they apply to this evaluation.

a) Identify relevant stakeholders and their concerns for the architecture(s) being evaluated.
b) Define value assessment criteria that contribute to key success factors, key indicators, and decisions that need to be made.
c) Define architecture analysis criteria that support the value assessment criteria.
d) Determine value assessment and architecture analysis criteria structure and relationships.
e) Determine relationship between value assessment and architecture analysis criteria and elements of value or utility (e.g. value function, utility curve).
   NOTE This task will determine how the two sets of criteria map to the “value curves” that represent the figures of merit for the architecture.
f) Select or develop analysis methods that support the defined value assessment and architecture analysis criteria
   NOTE These analysis methods could be in addition to or a refinement of the evaluation methods identified earlier.
g) Review assessment and analysis criteria and associated scales and weights (if any) with sponsor and architect.
h) Identify sources of information for use during application of the value assessment and architecture analysis criteria.

NOTE Some information will come from analysis, but other information could come from other sources, such as prior evaluation efforts, operational experience, industry databases, system verification activities, and research activities.

6.4.3.4 Establish evaluation techniques, methods and tools

a) Define analysis scales for measuring against the analysis criteria, if appropriate.
b) Define assessment scales for measuring against the assessment criteria, if appropriate.
c) Specify weights for assessment and analysis criteria, if appropriate.
NOTE Some methods do not use weights while others depend on them to achieve more accurate results. There are different kinds of weights, such as importance weights, swing weights, criticality weights. The methods chosen will usually specify the kinds of weights to be used.
d) Determine where on these scales the architecture is now and identify desired point(s) for future levels of achievement.
e) Identify relevant architecture attributes and associated measures.
f) Identify metrics to be determined from the measures.
NOTE There is overlap between measures and metrics. Both can be qualitative or quantitative, but what distinguishes them is important. Measures are concrete, usually measure one thing, and are quantitative in nature (e.g. I have five apples). Metrics describe a quality and require a measurement baseline (I have five more apples than I did yesterday). Measures and metrics can be useful for setting program priorities, allocating resources, and measuring performance. See ISO/IEC 15939 Measurement process.
g) Define relationship between measures, metrics and evaluation criteria.
h) Identify sources of information for obtaining values for these measures and metrics.
i) Identify techniques, methods and tools appropriate for these measures, metrics and evaluation criteria.

6.4.3.5 Review evaluation-related information

a) Identify relevant information for the chosen architecture analysis methods.
NOTE If possible, reuse existing data and results from previous evaluations of this kind, if the information available is still valid.
b) Collect all relevant and necessary information, including required architecture views and models.
c) Create additional information if not readily obtainable (i.e., non-existent, inaccessible) and if its creation is feasible within the available time without causing disruptions.
d) Examine and qualify collected artefacts in terms of completeness, correctness and consistency.
e) Develop an understanding of the architecture, the architecture quality attributes, key decisions and system concerns.
NOTE Quality attributes can be defined to characterize how stakeholder concerns could be addressed.

6.4.3.6 Analyze architecture attributes and assess stakeholder satisfaction

a) Identify architecture alternatives for evaluation, if appropriate.
NOTE 1 The alternatives may come from the architecture conceptualisation or elaboration activities.
NOTE 2 The status quo is sometimes one of the alternatives to be considered.
NOTE 3 Development of these alternatives may be outside the scope of the evaluation activity. However, sometimes the evaluation activity determines that there is an insufficient number, variety,

or extent of alternatives that have been predefined and the additional alternatives need to be generated.

b) Use the assessment and analysis method(s) as specified in the evaluation work plan to assess architecture for the identified purpose.
c) Use selected evaluation methods to determine concepts and properties of the architecture (alternatives) with respect to the evaluation criteria.

EXAMPLE Examples of methods to determine these concepts and properties include elements such as analysis, observation, simulation, prototyping, experimentation, inspection, audit, review, walk-through, and expert judgment.

d) Identify and characterize trade-offs.
e) Ensure relevant mandates and imperatives are met by the architecture(s).
f) Review analysis and assessment results with the sponsor, architect and other interested parties.

6.4.3.7 Formulate findings and recommendations

a) Identify and characterize findings from the evaluation.
b) Analyse the findings.
c) Validate the findings with subject matter experts and other relevant parties.
d) Assess implications of findings.
e) Develop recommendations.
f) Review findings and recommendations with the sponsor, architect and other interested parties.

6.4.3.8 Document and communicate evaluation results

a) Identify the audience for communicating the evaluation results.
b) Select most relevant and elaborate on key findings and recommendations.
c) Develop evaluation report in accordance with ISO/IEC 42030.
d) Obtain approval for report, if appropriate.
e) Present findings and recommendations to decision makers, if relevant.
f) Present to key stakeholders and architects.
g) Capture responses from these presentations (e.g. issues, action items, risks, observations, perspectives).
h) Redo parts of evaluation, if required or requested.
i) Update report, if necessary, based on feedback from presentations.
j) Archive report & responses received during presentations.

6.4.4 Work products

The following work products shall be compliant with ISO/IEC 42030:

a) Architecture evaluation plan,
b) Architecture evaluation report,
c) Architecture value assessment results,
d) Architecture attribute analysis results.

The following work products shall be stored in the organization-designated architecture repository:

a) Architecture evaluation plan,
b) Architecture evaluation report,
c) Architecture value assessment results,
d) Architecture attribute analysis results.

6.5 Architecture elaboration process

NOTE Architecture Elaboration process can be applied to current entities (to articulate and expound what the current architecture is), to future envisaged entities (to define the architectural bases on which they are formed), and to envisioned entities (to set architectural 'headmarks' against which future envisaged entities should make progress through the inclusion of relevant features and/or the provision of evolutionary capabilities).

6.5.1 Purpose

The purpose of the Architecture Elaboration process is to create one or more architecture descriptions in a form that uses established notations and languages and captures this in a set of consistent views and models.

NOTE 1 The ISO/IEC/IEEE 15288 standard specifies activities for dealing with systems that are designed and built by projects and programs within an organization. These activities are not usually appropriate for engineering the enterprise itself. Therefore, the Architecture Elaboration process will handle both system architectures (in the sense used in ISO/IEC/IEEE 15288) and enterprise architectures (in the sense used in ISO 15704).

NOTE 2 The views in an architecture description can be generated in any of the architecture processes. However, the more complete and refined set of views are generally created in the Architecture Elaboration process. These views created in the Architecture Elaboration process are usually composed of architecture models that use some kind of standardized modeling language or notation. ISO/IEC/IEEE 42010 provides requirements and guidance on how to create an architecture description.

6.5.2 Outcomes

An implementation of the Architecture Elaboration process shall achieve the following outcomes:

a) Architecture expressions are formalized using standardized or established architecture modeling languages and notations.
b) Architecture views and models accurately capture the fundamental concepts and properties of the architecture.
c) Alignment of the architecture with relevant requirements and design characteristics is achieved.

6.5.3 Activities and tasks

The organization or project shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the Architecture Elaboration process.

6.5.3.1 Prepare for architecture elaboration

a) Identify the intended users of the architecture description to be generated by this elaboration effort.
b) Identify the potential question(s) that can be addressed by the architecture elaboration effort.
c) Define the expected purpose, scope, objectives, and level of detail of the architecture elaboration effort.
d) Define an architecture elaboration strategy that is consistent with the architecture governance and management directives and guidance and is consistent with the purpose, scope and objectives.
e) Select or develop the requisite architecture elaboration methods and tools.
f) Plan the architecture elaboration effort.
g) Establish metrics for the architecture elaboration effort.
h) Collect the data and information needed for the architecture elaboration effort.
i) Obtain access to enablers needed for the architecture elaboration effort.

NOTE The enablers will usually be obtained from the Architecture Enablement process. When enablers are obtained from elsewhere, these can become candidate enablers for use by other projects through the Architecture Enablement process.

6.5.3.2 Manage architecture elaboration activities

a) Report architecture elaboration activity plans and status.
b) Monitor and assess whether architecture governance directives and guidance are being followed.
c) Monitor and assess whether architecture management directives and guidance are being followed.
d) Monitor and assess metrics for the architecture elaboration effort.
e) Identify and assess risks and opportunities associated with the architecture elaboration effort.
f) Maintain traceability of architecture elaboration results to the source material used during the process.
g) Organize, assess and control evolution of the architecture models and views.
h) Ensure the architecture description is maintained.
i) Provide key information items that have been selected for baselines.
j) Ensure personnel are trained in the use of identified techniques, methods and tools.
k) Ensure that relevant technical, project and organizational processes are properly using architecture elaboration products.
NOTE See section 7.1 for interactions with system life cycle processes.
l) Ensure that relevant enterprise processes are properly using architecture elaboration products.
NOTE See section 7.2 for interactions with enterprise life cycle processes.

6.5.3.3 Develop architecture viewpoints

a) Select, adapt, or develop viewpoints and model kinds based on stakeholder concerns.
NOTE ISO/IEC/IEEE 42010 can be used to assist in developing viewpoints.
b) Establish or identify potential architecture framework(s) to be used in developing models and views.
c) Capture rationale for selection of framework(s), viewpoints and model types.
d) Define purpose and scope of each model and view to be developed.
e) Select or develop supporting modeling methods and tools.

6.5.3.4 Develop models and views of the architecture(s)

a) Define the architectural context and boundaries in terms of interfaces and interactions with external entities.
b) Identify architectural entities and relationships between entities that address key stakeholder concerns and address architecture objectives.
NOTE Entities in the architecture are things that exist as particular and discrete units. They are neither necessarily tangible objects nor visible items. Examples of architecture entities includes such things as organizations, facilities, activities, roles, personnel, techniques, processes, policies, rules, principles, objectives, capabilities, nodes, links, system elements, interfaces, data elements, layers, protocols, hardware items, software items, etc.

c) Allocate relevant concepts, properties, characteristics, behaviors, functions, features, or constraints to architectural entities and relationships.
d) Select, adapt, or develop models of the architecture.
e) Compose views from the models in accordance with identified viewpoints to express how the architecture addresses stakeholder concerns and architecture objectives, and meets stakeholder and system requirements.
f) Harmonize the architecture models and views with each other.
g) Generate or modify the architecture description based on relevant models and views.

6.5.3.5 Relate the architecture to design

a) Identify design elements that relate to architectural entities and the nature of these relationships.
b) Define the interfaces and interactions between the design elements and with external entities.
c) Partition, align and allocate requirements to architectural entities and design elements.

d) Map design elements and architectural entities to design characteristics.
e) Formalize principles for the design and evolution of the entity being architected.

6.5.3.6 Assess the architecture elaboration

a) Assess each architecture view and model against architecture elaboration objectives, purposes and questions.
b) Assess the architecture description to determine if it meets the needs of intended users.
c) Assess the architecture description for consistency, completeness and correctness.
d) Update the architecture description to address identified gaps and shortfalls.
e) Place architecture description in the architecture repository.
f) Identify views and models that can be generalized for reuse by other projects.

6.5.4 Work products

The following work products shall be compliant with ISO/IEC/IEEE 42010:

a) Architecture viewpoints,
b) Architecture views,
c) Architecture models,
d) Architecture descriptions.

The following work products shall be stored in the organization-designated architecture repository:

a) Architecture viewpoints,
b) Architecture views,
c) Architecture models,
d) Architecture descriptions.

6.6 Architecture enablement process

6.6.1 Purpose

The purpose of the Architecture Enablement process is to ensure that the supporting capabilities needed to perform the architecture activities in the other architecture processes are available when and where necessary. This could involve the acquisition or development of enabling tools and methods, frameworks and templates, reference architectures and process documentation, methodologies and modelling languages. This process also provides the necessary training on how to use these supporting capabilities.

This process also develops and maintains key support capabilities such as the architecture repository, reference architectures, modelling tool licensing and certification, process documentation, configuration management, etc.

Architecture frameworks and viewpoints are often key enablers for development of architecture descriptions. Refer to ISO/IEC/IEEE 42010 for requirements and guidance on developing architecture descriptions, architecture viewpoints, and architecture frameworks.

Architecture Enablement process activities provide services, means and data to sustain the other architecture processes.

6.6.2 Outcomes

An implementation of the Architecture Enablement process shall achieve the following outcomes:

a) A set of tools, methods and processes to enable architecting.

b) A set of enablers that facilitate architecting.

c) Information and information flows for architecture enablement.

6.6.3 Activities and tasks

The organization or project shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the Architecture Enablement process.

6.6.3.1 Prepare for architecture enablement

a) Identify the potential problem areas that need to be addressed by architecture enablement.
b) Define the scope and objectives of the architecture enablement capabilities.
c) Define an architecture enablement strategy that is consistent with the architecture governance and management directives and guidance.
d) Identify the potential enablement needs and opportunities for individual architectures, architecture alternatives, and portfolios of architectures.
e) Define the level of details of the architecture enablement effort.
f) Identify, the requisite architecture enablement methods and tools.
g) Plan the architecture enablement effort.
h) Establish metrics for the architecture enablement effort.
i) Collect the data and information and obtain access to enablers needed for the architecture enablement effort.

6.6.3.2 Manage architecture enablement activities

.

a) Report architecture enablement activity plans and status.
b) Monitor and assess whether architecture governance directives and guidance for architecture enablement are being followed.
c) Monitor and assess whether architecture management directives and guidance for architecture enablement are being followed.
d) Monitor and assess metrics for the architecture enablement effort.
e) Identify the enablement issues arising from changes to architecture.
NOTE For example, moving from a functional-based to a service-oriented approach can lead to significant change in the architecture enablement items.
f) Direct the evolution of enablement capabilities.
g) Identify and assess risks and opportunities associated with the architecture enablement effort.
h) Ensure architecture enablers are utilized properly.
i) Ensure enterprise processes are properly using architecture enablement capabilities.
j) Establish and maintain a catalog of the enablers that can be used for other architecture processes.
k) Develop necessary control and communication plans for architecture enablement.

6.6.3.3 Identify necessary architecture enablers

a) Define the requirements for capturing, communicating and sharing information about architecture.
b) Identify assets, competencies and capabilities that can be leveraged for driving the architectures vision, strategy, goals and objectives.
c) Identify roles and responsibilities of people who would facilitate architecture enablement.
d) Identify areas of improvement and additional training to be provided in order to drive the architecture vision, strategy, goals and objectives.
e) Identify services, activities, events and controls that can be applied for architecture enablement.
NOTE Events are milestones, reviews, audits, key decision points, quality gates, etc. Controls are checklist, entry criteria, exit criteria, decision trees, etc.
f) Identify information and information flows that can be used for architecture enablement.

.

6.6.3.4 Develop necessary architecture enablers

a) Select or develop the appropriate organizational resources, assets, activities and services for provision of architecture enablement.

b) Select or develop the infrastructure, tools and methods for architecture enablement.

NOTE Architecture frameworks and viewpoints are often key enablers for development of architecture descriptions. Refer to ISO/IEC/IEEE 42010 for requirements and guidance on developing architecture descriptions, architecture viewpoints, and architecture frameworks.

c) Select or develop information structures and information flows necessary for architecture enablement.

d) Select or develop assets, competencies and capabilities that can be leveraged for driving the architectures vision, strategy, goals and objectives.

e) Select or develop roles and responsibilities of people who would facilitate architecture enablement.

f) Select or develop areas of improvement and additional training to be provided in order to drive the architecture vision, strategy, goals and objectives.

g) Select or develop services, activities, events and controls that can be applied for architecture enablement.

NOTE Events are milestones, reviews, audits, key decision points, quality gates, Etc. Controls are checklist, entry criteria, exit criteria, decision trees, Etc.

6.6.3.5 Deploy capabilities for architecture enablement

a) Align activities, services, resources, assets and information cohesively for provision of effective architecture enablement.

b) Deploy the appropriate organizational resources, assets, activities and services for provision of architecture enablement.

c) Deploy the infrastructure, tools and methods for architecture enablement.

d) Deploy information structures and information flows necessary for architecture enablement.

e) Ensure that the right information reaches the right role players at the right time and place.

f) Understand the architecture dynamics and deploy appropriate capabilities for architecture enablement.

g) Understand gaps in enablement that caused issues in the past and look at ways and means of addressing these gaps.

.

6.6.4 Work products

Architecture frameworks shall be compliant with ISO/IEC/IEEE 42010.

Architecture frameworks shall be stored in the organization-designated architecture repository.

Architecture viewpoints shall be compliant with ISO/IEC/IEEE 42010.

Architecture viewpoints shall be stored in the organization-designated architecture repository.

Architecture data-based enablers shall be stored in the organization-designated architecture repository.

Architecture non-data-based enablers shall be registered in the organization-designated architecture repository.

7 External interactions

7.1 Relationship with system life cycle processes

The following table describes how architecture should be used during each stage of a system life cycle. For this description the stages are those described in the INCOSE SE Handbook and ISO 24748-1, figure 6:

Stages	Architecture usage
Concept	Selling the program, procurement of funding, discovery of needs with stakeholders,
Development	Requirements definition, system analysis, system design, etc.
Production	Problem resolution, production planning, integration planning
Utilisation	Operational planning, training of users, logistics planning, defect resolution
Support	Problem resolution, anomaly investigation, evolution planning
Retirement	Reuse planning, decommissioning decision, repurposing analysis

Figure 3 : Architecture use along a system life cycle

The architecture processes will interact with the following system life cycle processes (specified in ISO/IEC/IEEE 15288):

- Agreement processes
- Organizational project-enabling processes
- Technical management processes
- Technical processes

The links are identified in three cases:

- When system life cycle processes provide information to architecture-related processes. For example, the validation process might identify interfaces required for validation and fulfilment of constraints related to the architecture.
- When architecture processes provide information to a system life-cycle process. For example, the architecture definition process might provide a validated architecture description package to the design definition process.
- When the architecture process is implemented inside a particular system life cycle process.

Annex B specifies how system life cycle processes in ISO/IEC/IEEE 15288 should use architecture-related information. Annex B also specifies how architecture processes should use information items from 15288 processes.

7.2 Relationship with enterprise processes

The following table describes how architecture should be used during each stage or phase of life cycle of an enterprise or any of its entities. For this description the stages are those described in the figure A.2 – GERA life-cycle phases for any enterprise or entity of the ISO 15704:

Stages	Architecture usage
Identification	Selling the venture, procurement of funding,
Concept	Discovery of needs with stakeholders, identification of the enterprise assets. NOTE Architecture may be used to structure or restructure the enterprise itself e.g. conceptualisation.
Requirements	Finding and definition of the true-requirements, identification of the enterprise projects
Design	Enterprise analysis, preliminary and detailed design, definition of the enterprise projects
Implementation	Problem resolution, production planning, integration planning, set up of the enterprise projects
Operation	Operational planning, training, logistics planning, monitoring of the enterprise projects, anomaly investigation, problem resolution, evolution planning
Decommissioning	Reuse planning, decommissioning decision, repurposing analysis

Figure 4 : Architecture use along an enterprise life cycle

The architecture processes will interact with the enterprise engineering process (specified in ISO 15704 A.3.2 – Enterprise engineering methodologies), taking into account aspects like:

- Human factors
- Project management
- Economic, financial and commercial considerations.

The links are identified in three cases:

- When enterprise life cycle processes provide information to architecture-related processes. For example, when the enterprise strategy impacts the architecture governance activities.
- When architecture processes provide information to the enterprise life cycle process. For example, the architecture definition process updates the vision of the asset management of the enterprise.
- When the architecture process is implemented inside a particular enterprise life cycle process. For example, when the architecture definition activities are perform to elaborate an enterprise architecture aiming to structure the enterprise itself and its projects.

8 Architecture-related work products

NOTE The following are requirements that generally apply to all architecture process-generated work products. Requirements on work products for a particular process are specified in the clause for that process.

Architecture work products shall conform to organization-specific templates, where applicable.

Architecture work product templates shall be retained in the organization-designated architecture repository.

Finished architecture work products shall be stored in the organization-designated architecture repository.

Annex A (normative) Tailoring process

NOTE This annex is an adaptation of annex A in ISO/IEC/IEEE 15288.

A.1 Introduction

This Annex provides requirements for the tailoring of this International Standard.

NOTE 1 Tailoring is not a requirement for conformance to the standard. In fact, tailoring is not permitted if a claim of "full conformance" is to be made. If a claim of "tailored conformance" is made, then this process is applied to perform the tailoring.

NOTE 2 Additional guidance for tailoring can be found in the ISO/IEC/IEEE TR 24748 guides, on the application of life cycle processes.

A.2 Tailoring process steps

A.2.1 Purpose

The purpose of the Tailoring process is to adapt the processes of this International Standard to satisfy particular circumstances or factors that:

1) Surround an organization that is employing this International Standard in an agreement;
2) Influence a project that is required to meet an agreement in which this International Standard is referenced;
3) Reflect the needs of an organization in order to supply products or services.

A.2.2 Outcomes

As a result of the successful implementation of the tailoring process:

1) Modified or new life cycle processes are defined to achieve the purposes and outcomes of a life cycle model.

A.2.3 Activities and tasks

If this International Standard is tailored, then the organization or project shall implement the following tasks in accordance with applicable policies and procedures with respect to the Tailoring process, as required.

1) Identify and record the circumstances that influence tailoring. These influences include, but are not limited to:
   a) stability of, and variety in, operational environments;
   b) risks, commercial or performance, to the concern of interested parties;
   c) novelty, size and complexity;
   d) starting date and duration of utilization;
   e) integrity issues such as safety, security, privacy, usability, availability;
   f) emerging technology opportunities;
   g) profile of budget and organizational resources available;
   h) availability of the services of enabling systems;
   i) roles, responsibilities, accountabilities and authorities in the overall life cycle of the system;
   j) the need to conform to other standards.

2) In the case of properties critical to the system, take due account of the life cycle structures recommended or mandated by standards relevant to the dimension of the criticality.
3) Obtain input from parties affected by the tailoring decisions. This includes, but may not be limited to:
    a) the system stakeholders;
    b) the interested parties to an agreement made by the organization;
    c) the contributing organizational functions.
4) Make tailoring decisions in accordance with the decision management process to achieve the purposes and outcomes of the selected life cycle model.

    NOTE 1 Organizations establish standard life cycle models as a part of the Life Cycle Model management process. It is sometimes appropriate for an organization to tailor processes of this International Standard in order to achieve the purposes and outcomes of the stages of a life cycle model to be established.

    NOTE 2 Projects select an organizationally-established life cycle model for the project as a part of the Project Planning process. It is sometimes appropriate to tailor organizationally adopted processes to achieve the purposes and outcomes of the stages of the selected life cycle model.

    NOTE 3 In cases where projects are directly applying this International Standard, it is sometimes appropriate to tailor processes of this International Standard in order to achieve the purposes and outcomes of the stages of a suitable life cycle model.

5) Select the life cycle processes that require tailoring and delete selected outcomes, activities, or tasks.

    NOTE 1 Irrespective of tailoring, organizations and projects are always permitted to implement processes that achieve additional outcomes or implement additional activities and tasks beyond those required for conformance to this standard.

    NOTE 2 An organization or project sometimes encounter a situation where there is the desire to modify a provision of this International Standard. Modification is to be avoided because of unanticipated consequences on other processes, outcomes, activities or tasks. If necessary, modification is performed by deleting the provision (making the appropriate claim of tailored conformance) and, with careful consideration of consequences, implementing a process that achieves additional outcomes or performs additional activities and tasks beyond those of the tailored standard.

Annex B (normative) Interactions with system life cycle processes

System life cycle processes in ISO/IEC/IEEE 15288 should use architecture-related information as specified in Table 2.

Table 2: Uses of architecture by 15288 processes

15288 Clause	System Life Cycle Process	Uses of Architecture by this Process
6.1	**Agreement processes**	
6.1.1	Acquisition	Basis of supplier evaluation
6.1.2	Supply	Basis of solution to be supplied
6.2	**Organizational project-enabling processes**	
6.2.1	Life cycle model management	Identification of systems and system elements, system transition points
6.2.2	Infrastructure management	Identification of infrastructure needed by systems
6.2.3	Portfolio management	Identification of systems and system elements, system transition points, system inter-dependencies
6.2.4	Human resource management	Determination of necessary knowledge, skills and expertise
6.2.5	Quality management	Identification of systems and system elements
6.2.6	Knowledge management	Identification of architecture features to be used for tagging information items in knowledge repository, management of architectures and architecture information
6.3	**Technical management processes**	
6.3.1	Project planning	Identification of systems and system elements, system transition points, system attributes and measure, system inter-dependencies
6.3.2	Project assessment and control	Identification of systems and system elements, system transition points, system attributes and measures
6.3.3	Decision management	Architecture evaluation recommendations. Identification of systems and system elements, system transition points, system attributes and measures.
6.3.4	Risk management	Identification of systems and system elements, system transition points, system attributes and measures, system inter-dependencies
6.3.5	Configuration management	Identification of systems and system elements, system interfaces, system configurations and options
6.3.6	Information management	Identification of architecture features to be used for tagging information items to be managed
6.3.7	Measurement	Identification of systems and system elements, system transition points, system attributes and measures
6.3.8	Quality assurance	Identification of systems and system elements, system transition points, system attributes and measures
6.4	**Technical processes**	
6.4.1	Business or mission analysis	Understanding of current and planned architectures and related systems
6.4.2	Stakeholder needs and requirements definition	Understanding of current and planned architectures and related systems. Identification of architecture features and functions.
6.4.3	System requirements definition	Identification of systems and system elements, system transition points, system attributes and measures. Understanding of current and planned architectures and related systems. Identification of architecture features and functions.
6.4.4	Architecture	Basis for definition of architecture, architecture conceptualisation,

15288 Clause	System Life Cycle Process	Uses of Architecture by this Process
	definition	architecture elaboration
6.4.5	Design definition	Basis for design of system and non-system solutions. Note: Design definition may employ architecture at lower levels within a system hierarchy, e.g., at the system element level.
6.4.6	System analysis	Identification of systems and system elements, system transition points, system attributes and measures. Understanding of current and planned architectures and related systems. Identification of architecture features and functions. Note: Problem analysis will occur as part of the Architecture Conceptualization process.
6.4.7	Implementation	Understanding of intended use of architecture-related systems.
6.4.8	Integration	Identification of systems and system elements, system transition points, system attributes and measures. Understanding of current and planned architectures and related systems. Identification of architecture features and functions. Note: Integration addresses the composition of systems from their constituent elements and the integration of systems into their operational context/environment. Such considerations are identified, conceptualised and elaborated by architecting. So, architecting can be applied for this purpose.
6.4.9	Verification	Identification of systems and system elements, system transition points, system attributes and measures. Understanding of current and planned architectures and related systems. Identification of architecture features and functions.
6.4.10	Transition	Identification of systems and system elements, system transition points, system attributes and measures. Understanding of current and planned architectures and related systems. Identification of architecture features and functions.
6.4.11	Validation	Identification of systems and system elements, system transition points, system attributes and measures. Identification of architecture features and functions.
6.4.12	Operation	Identification of systems and system elements, system transition points, system attributes and measures. Understanding of current and planned architectures and related systems. Identification of architecture features and functions.
6.4.13	Maintenance	Identification of systems and system elements, system transition points, system attributes and measures. Understanding of current and planned architectures and related systems. Identification of architecture features and functions.
6.4.14	Disposal	Identification of systems and system elements, system transition points. Understanding of current and planned architectures and related systems. .

Architecture processes in ISO/IEC 42020 should use system life cycle-related information as specified in Table 3.

Table 3: Information used by 42020 architecture processes

15288 Clause	System Life Cycle Process	Information Used by Architecture Processes
6.1	**Agreement processes**	
6.1.1	Acquisition	Acquisition plans, supplier evaluation criteria
6.1.2	Supply	Proposed solution attributes
6.2	**Organizational project-enabling processes**	
6.2.1	Life cycle model management	Life cycle models of systems and system elements
6.2.2	Infrastructure management	Infrastructure features, functions and services
6.2.3	Portfolio management	Portfolio evaluation criteria, program and project dependencies
6.2.4	Human resource management	Knowledge, skills and expertise of current or planned personnel
6.2.5	Quality management	Quality assessment criteria
6.2.6	Knowledge management	General information from knowledge repository
6.3	**Technical management processes**	
6.3.1	Project planning	Project plans
6.3.2	Project assessment and control	Project assessment data
6.3.3	Decision management	Architecture-related decisions, decision criteria
6.3.4	Risk management	Identification of system risks and risk mitigation plans
6.3.5	Configuration management	Identification of configuration items. Baselined requirements and requirements changes (proposed and actual).
6.3.6	Information management	General information from information repository
6.3.7	Measurement	Measurement parameters and values
6.3.8	Quality assurance	Quality assurance criteria
6.4	**Technical processes**	
6.4.1	Business or mission analysis	Business needs, gaps and shortfalls. Mission needs, gaps and shortfalls.
6.4.2	Stakeholder needs and requirements definition	Identification of stakeholders and their concerns. Prioritization of concerns. Definition of perceived needs, gaps and shortfalls.
6.4.3	System requirements definition	Proposed and approved system requirements. Identification of key requirements constraints, conditions and challenges.
6.4.4	Architecture definition	Architecture plans and roadmaps. Architecture descriptions.
6.4.5	Design definition	Design plans and roadmaps. Design descriptions. Design evaluation results.
6.4.6	System analysis	System analysis results. System analysis tools methods capabilities and limitations.
6.4.7	Implementation	Implementation plans and roadmaps. Identification of key implementation constraints, conditions and challenges.

15288 Clause	System Life Cycle Process	Information Used by Architecture Processes
6.4.8	Integration	Integration plans and roadmaps. Identification of key integration constraints, conditions and challenges.
6.4.9	Verification	Verification plans and roadmaps. Identification of key verification constraints, conditions and challenges. Note: Verification approaches may impose issues such as architecture controllability and observability.
6.4.10	Transition	Transition plans and roadmaps. Identification of key transition constraints, conditions and challenges.
6.4.11	Validation	Validation plans and roadmaps. Identification of key validation constraints, conditions and challenges.
6.4.12	Operation	Operations plans and roadmaps. Identification of key operations constraints, conditions and challenges.
6.4.13	Maintenance	Maintenance plans and roadmaps. Identification of key maintenance constraints, conditions and challenges.
6.4.14	Disposal	Disposal plans and roadmaps. Identification of key disposal constraints, conditions and challenges.

Annex C (informative) Relationship with other standards

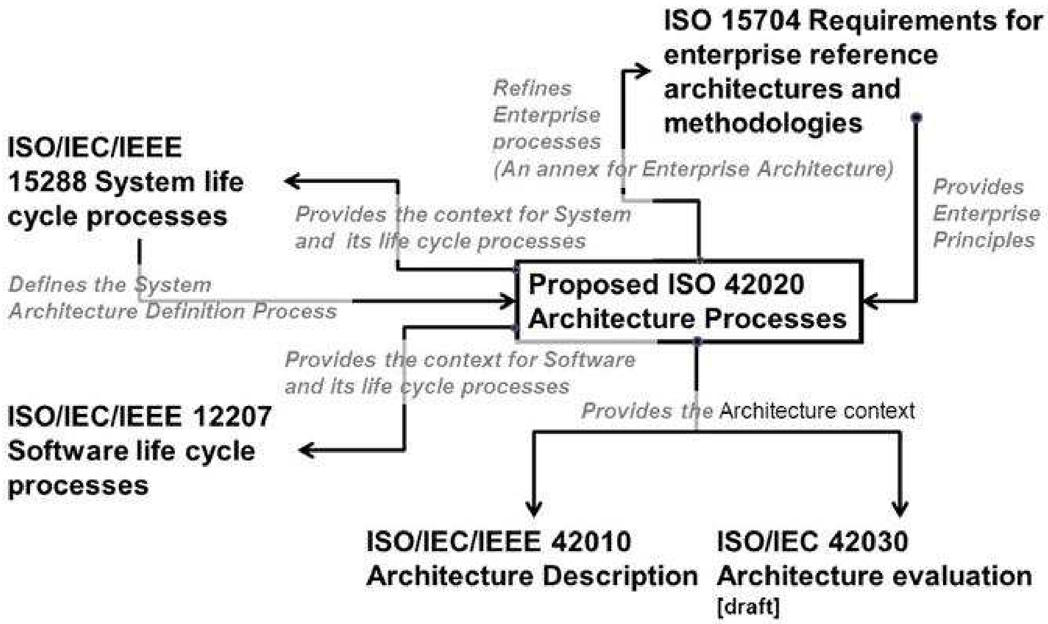
The following list provides standards related to architecture:

- ISO/IEC/IEEE 42010 Systems and software engineering — Architecture description.
- ISO/IEC 42030 Systems and software engineering — Architecture Evaluation.
- ISO/IEC/IEEE 15288 Systems and software engineering — System life cycle processes.
- ISO/IEC/IEEE 15289 Systems and software engineering — Content of life-cycle information products (documentation)
- ISO/IEC/IEEE TR 24748-1 Systems and software engineering — Life cycle management.
- ISO/IEC/IEEE 12207 Systems and software engineering — Software life cycle processes.
- ISO 15704 Industrial automation systems — Requirements for enterprise-reference architectures and methodologies.
- ISO/CEN 19439 Enterprise integration — Framework for enterprise modelling.
- ISO/IEC 38500 – Corporate governance of information technology.
- ISO/IEC TR 38502:2014 Information technology — Governance of IT — Framework and models.

NOTE The JTC1/SC7 Architecting Guidance Study Report provides the main references about architecture and architecting.

The following figure describes the main relationships between this International Standard and other ISO standards related to architecture and related activities. This international standard does the following:

- Refines the Architecture Definition process of the ISO/IEC/IEEE 15288 and ISO/IEC/IEEE 12207
- Frames the processes of these two standards
- Refines the Enterprise Reference Architecture of the ISO 15704 and the process description provided in the GERAM annex.
- Is considered with the enterprise principles defined by ISO 15704
- Provides the architecture context where the ISO/IEC/IEEE 42010 and ISO/IEC 42030 serve to respectively describe Architecture Description and Architecture Evaluation.



1736

1737 **Figure 5: Main relationships between ISO/IEC 42020 and other ISO standards**

1738

Annex D (informative) Notes on terms and concepts

This annex complements Clause 5 with additional information about key terms and concepts used in this standard.

D.1 Architecting scenarios and approaches

D.1.1 Architecting scenarios

An architecting scenario defines the starting point for conducting an architecting activity. It defines the background and reasons for trying to form a system concept. While there is no complete list, there are a number of common cases. The scenario, the approach, and the other points made here are distinct but not independent. Arbitrary combinations will often not make practical sense, but neither does each solely depend on the other.

Note that these scenarios are described in terms of being applied to a “system” architecture although they can be also applied to enterprise architectures, as well as to products and services that are not otherwise considered to be systems (e.g., software item).

- New development, (sometimes known as a “greenfield” approach)

  This is the classic case where there is no prior system, the system being architected will be all new. In the most extreme case the sponsor will desire a system delivering an unprecedented capability implemented with a technology with limited or non-existent precedent (consider the development of nuclear submarines in the 1950’s).

- New Product in a Product-Line

  In this case there are pre-existing systems with capabilities and technology quite similar to the desired system. The new system is intended to make a precedented extension of the pre-existing family-of-systems.

- Legacy evolution

  Here the architect has an existing system or collection of systems (the “legacy”) and is tasked to evolve or extend that legacy. The goals may be to add capability, reduce operating cost, eliminate obsolete technology, or something else. A key aspect is that the legacy exists and cannot be abandoned.

- Legacy revolution

  In this case the legacy exists but the sponsors intent is to radically depart from that legacy. The usual case here is that while the legacy works there is a belief that the adoption of a radical departure (usually a radical departure in both technical approach and concept of operations) will enable large changes in capabilities or costs. The situation is not entirely a new development because the legacy capability exists and the “revolutionary” system will presumably have to interface to it in some fashion, but the deliberate intent is to abandon some large fraction of existing infrastructure.

- Incremental start

  Here the sponsor wants to build something new, but the uncertainty about what will be most fit is great. Instead of making a large-scale and irrevocable bet on the future system the intent is to make an incremental step, to be followed by later steps that move toward a superior system. The architecting activity much embraces the uncertainty and develop both initial steps and options for subsequent steps that account for known (and possibly unknown) uncertainties in user demand or technology.

- Product-line start

  This is a particular case of a new start where the intent is to build a product-line or family-of-systems not a single system. The result should be the infrastructure used to produce an indeterminate number of future, related systems, related by reliance on a common base of design, technology, or production.

D.1.2 Architecting approaches

Several different approaches to architecting (i.e. conceptualisation, evaluation and elaboration) exist. These are categorised primarily in terms of the strategy (starting points) for forming the system architecture. The approach which should be adopted depends upon the complexity of the system of interest, its novelty, realisation mechanisms, and/or the uncertainty in the stakeholder needs.

Note that a particular system architecting effort sometimes requires the use of more than one of these approaches.

Some examples of architecting approaches include:

- Bottom-up

  In bottom-up architecting the starting points are the artifacts, capabilities and/or services that are available and/or realisable which are then composed and formed into a system architecture exhibiting the desired (or desirable) emergent properties.

  Note that in studying an existing system there is a clear distinction between producing an architecture description document and comprehending the pre-existing architecture of the system. One may have a clear comprehension of the architecture of the system of interest without having developed a complete document. Conversely, one may have a comprehensive document (albeit one badly written) and still have no clear comprehension of any organizing structure or principles of the system (perhaps because there are no such organizing abstractions).

- Middle-out

  In the middle-out approach an arbitrary level of abstraction in the system hierarchy is used as the starting point. Reasoning about, and architecting of, the system then progresses both upwards (towards the goals) and downwards (towards artifacts/capabilities/services).

- Outer-in

  This approach to system architecting starts at both the top (system goals) and bottom (artifacts/capabilities/services) and works towards the middle. It entails balancing and harmonising desirable and achievable system properties.

- Reverse

  Reverse architecting is aspect of reverse engineering for making the architectures of existing (or designed) systems explicit. It involves the extraction, abstraction and presentation of system information. The devising of “as is” architectures can be devised in this manner. Reverse architecting may address the goals of the existing systems, their components, or both.

- Top-down

In top-down system architecting the starting points are the system goals. The approach proceeds through conceptualisation to form the system architecture, stopping when appropriate levels of definitional formality and detail have been achieved. The devising of “to-be” architectures generally involves some top-down architecting.

D.2 Architecting kinds and styles

D.2.1 Architecture kinds

Different kinds of architecture can be considered according to their purpose, domains of application, and roles within the system and architecture life cycles. System architecting may require the use (including development and/or application) of architectures of several kinds.

NOTE Service-Oriented Architecture (SOA) is sometimes identified as an architecture kind but is more accurately termed an architecture style. In fact, any of the architecture kinds considered can be defined with a service orientation.

In discussing architecture kinds it is important to keep in mind the distinction in ISO standards (especially ISO/IEC/IEEE 42010) between architecture descriptions (as documents) and architectures (as conceptual things).

Some commonly used architecture kinds include:

- Enterprise architecture

  An enterprise architecture is the architecture of an enterprise. [INCOSE UK APP] It has a defined overall business objective and may include one or more participating organization.

  Enterprise architecture is the organizing logic for business processes and IT infrastructure reflecting the integration and standardization requirements of the company's operating model. The operating model is the desired state of business process integration and business process standardization for delivering goods and services to customers. [MIT Center for Information Systems Research, Peter Weill, Director, as presented at the Sixth e-Business Conference, Barcelona Spain, March 27, 2007]

- Overarching architecture

  An overarching architecture provides a high-level architectural context for systems and their associated architectures, including the interactions between these systems and any dependencies between the architectures. It concentrates on high-level objectives at the level of capabilities, systems of systems, or portfolios of projects and of necessity addresses such considerations at a comparatively high level of abstraction given the breadth of coverage.

  NOTE Notion of overarching architecture is described in the NATO Architecture Framework and The Open Group Architecture Framework (TOGAF).

  EXAMPLE An overarching architecture can be done for maritime surveillance of a country. This architecture orients programs and projects which occur in this scope, focusing particular domains of activities, like maritime search and rescue.

- Reference architecture

  A reference architecture is used by a community of interest as a shared and agreed reference system description that can be used for that community's business purposes. It is usually generic and is instantiated as system architectures specific for individual business purposes. Reference architectures are used to (1) aid understanding of the forms of likely solutions to problems within a particular domain, and (2) to maximize the possible commonality in forms of solutions to similar problems within such a domain.

NOTE Notion of reference architecture is described in many architecture frameworks, engineering methodologies and guides.

EXAMPLE See OASIS Reference Architecture for Service Oriented Architecture

- Baseline architecture

  Baseline architecture is the definition of the architecture being defined for a given point of time. Particular cases are:

  - Current architecture (or “as-is” architecture) is the definition of the architecture currently in use
  - Target architecture (or “to-be” architecture) gives the expected definitive definition.

  NOTE A formal definition of target architecture is provided below.

- Target architecture

  Target architecture is a description of an envisioned future state of the architecture concerning the system of interest.

- Functional architecture

  ISO 42010 compliant architecture description document will normally contain a view capturing the functions of the system of interest (it is not a 42010 normative requirement but would almost always be included). A functional architecture can be said to be the essential or organizing functional structure, the way abstracted inputs are transformed into outputs by the system to achieve its mission.

- System Architecture

  System architecture addresses the architecture of a system. It can be defined for one or more epochs according to a roadmap. It can also be defined for different kinds of system (e.g. product line architecture, enterprise architecture) and at different levels of functional abstraction (e.g. logical architecture, physical architecture).

  NOTE 1 System is used here to cover anything studied with a systemic approach.

  NOTE 2 System architecture definition will apply the directives given by the relevant Overarching Architecture, if any.

  NOTE 3 A system architecture can be derived totally or partially from one or several reference architectures.

  NOTE 4 If the system architecture is defined for one or more epochs according to a roadmap, there is one (or several) system baseline architecture and one system target architecture.

- Logical architecture

  The logical view of an architecture in an architecture description is typically an integrated model of both the system’s functions and retained data, where the abstraction level is chosen to be directly relevant to users rather than implementation. The logical view of the architecture defines data as it makes sense in the problem domain, and defers definition of how it might be represented to other views. So, for example, the logical view would define data and functional transformations in terms of positions, currency amounts, or objects of user interest and not in terms of XML records or database fields.

- Physical architecture

  A physical view of the architecture description is an arrangement of system elements and physical interfaces which provides the design solution for a product, service, or enterprise, and is intended to satisfy logical architecture elements and system requirements. It is defined in terms of physically recognizable objects that compose a systems implementation and the physical interfaces among them. It is implementable through technologies. [ISO/IEC/IEEE DIS 15288: 2014]

- Product line architecture

  According to ISO-24765, a product line is:

  - From the commercial viewpoint, group of products or services sharing a common, managed set of features that satisfy specific needs of a selected market or mission.

  - From the engineering viewpoint, a collection of systems that are potentially derivable from a single domain architecture.

  Architecture description of product line formalizes a set of products addressing similar problems and exhibiting an appropriate degree of architectural and solution commonality.

- System of Systems Architecture

  The architecture of a system which meets the criteria for a “system-of-systems”. Systems of systems should be distinguished from large but monolithic systems by the independence of their components, their evolutionary nature, emergent behaviour, and a geographic extent that limits the interaction of their components to information exchange. [Mark Maier, “Architecting Principles for Systems-of-Systems”]

D.2.2 Architecture styles

An architecture (architectural) style is a set of principles and/or a generic pattern that provides an abstract framework for multiple systems. It is defined by the system architecture elements, their topological layout, connectors and interaction mechanisms, and applicable constraints. Architecture styles may describe deployment patterns, structure and design issues, and communication factors. Their use improves system structuring and understanding (through the use of established mechanisms and vocabularies), promotes design reuse (through the development of systems based upon proven forms of solution), and supports the consideration of pertinent technical issues.

NOTE Architecture styles are distinct from the notion of architecting styles. Architecting styles refer to ways of architecting which are codified according to the architecture’s primary purpose including its degree of influence. [See INCOSE UK APP or Wilkinson/Evans papers]

Some examples of architecture styles particularly as applicable to computer-based systems are:

- Client server

  This architecture distributes data and processing physically across different types of system element. Servers provide specific services such as printing, data management, etc. Clients call on these services. Networks allow clients to access servers.

- Component-based architecture

  This style decomposes system functionality into reusable cohesive functional or logical components that expose well-defined communication interfaces.

- Data-driven architecture

  Data-driven architectures are concerned with the acquisition, manipulation and dissemination of data. They may be considered as being composed of pipelines of filters (which perform functional transformations of input data to produce data output) and pipes (which convey streams of data).

- Event-driven architecture

  Event-driven architectures promote the production, detection, consumption of, and reaction to events, where an event is a significant change in system state. Event-driven systems comprise event emitters (or agents), event consumers (or sinks), and event channels.

- Layered architecture

  Layered Architectures hierarchically structure functionality as several layers of increasing abstraction typically ranging from a problem focus at the top level to realisation considerations at the lowest level. Interaction between layers is often restricted to adjacent layers.

- Object-oriented and Object-Request Broker (ORB)

  An object is a collection of functions (called methods) and associated data. Object-orientation is an analysis and design paradigm based on the division of responsibilities for a system into individual reusable and self-sufficient objects, each containing the data and the behaviour relevant to the object.

  An ORB manages interactions between distributed objects. It knows about the objects that are requesting services and handles object communications. The communicating objects do not need to know the location of other objects nor do they need to know anything about their implementation. [Ian Sommerville, “Software Engineering”, 10th edition?] Distributed object systems may be formed from objects from different suppliers with ORBs addressing their communication issues.

- Publish-subscribe

  This is a style of information exchange in which certain system elements offer data to other system elements through published messages. System elements requiring such input data subscribe to the relevant messages.

- Repository architecture

  In this architecture the information exchange between system elements is physically realised through a central data repository which can be accessed (and where appropriate, contributed to) by all system elements.

- Service-Oriented Architecture (SOA)

  SOA is an Architecture Style that supports the service-orientation paradigm by exposing (and consuming) functionality from distributed systems as independent services in the form of stateless functions and using contracts and messages. SOA promotes reuse at the macro (service level) rather than micro (e.g. object) level.

Any of the architecture kinds described above can be defined with a service orientation. With this approach, services are preferred to expressed outcomes and interactions between entities (actors and constituents of the solution), with consideration of various operational, system, applicative and technical views.

Systems can support multiple architecture styles (heterogeneous architecture) for example to support different architectural concerns (information centricity, process/flow-orientation) but this increases solution complexity. The use of a single architecture style (homogeneous architecture) may ease design but may compromise certain required (or desired) system capabilities

D.3 Architecture motivation model

Enterprise activities, including architecture ones, will be driven by a set of motivation elements:

- Business aspiration including vision, goal, objectives and mission: these elements
- Business means including strategy, policies, rules and guidance
- Business constraints including laws, regulation and influencers
- Existing and expected business assets including products, tools and people

EXAMPLE An example of an architecture motivation model is the OMG Business Motivation Model.

These motivation elements form a dashboard for the architecture activities:

- Aspiration elements are elaborated and used by governance
- Means are drivers for management
- Constraints and assets have to be considered for any process

Criteria are derived from the dashboard for analysis and assessment of both architectures and architectures activities.

D.4 Architecture quality attributes

Architecture quality attributes are the degree to which the architecture is able to deliver value to its stakeholders. It is a set of essential and distinguishing attributes that have a pragmatic interpretation of the architecture's inferiority or superiority. It is a function of: 1) Architecture process outcomes, 2) Impact of the architecture on various stakeholders, 3) Measure of degree of achievement of stakeholder concerns, 4) Measure of capabilities of the architecture.

Architecture quality attributes are the overall factors that affect behaviour, structure, design and experience of architectures. They represent areas of concern that potentially impact the spatial structure and behaviour exhibited by the realized system. The extent to which the architecture handles a combination of quality attributes indicates the success of the architectural design and overall quality of the realized system.

The taxonomy for each architecture quality attribute would be:

a) Concerns: The parameters by which the attributes are measured.

b) Factors: Policies and mechanisms of the system and its environment that impacts the stakeholder concerns.

c) Methods: Techniques for addressing concerns and processes for realizing the quality attributes during productions.

While conceptualizing architecture to address the architecture quality attributes, it is necessary to consider potential impact of each of the quality attributes on other stakeholder concerns. While trade-off analysis techniques aid architects in prioritizing architecture quality attributes, architectural tactics describe how a specific quality attribute can be achieved. The importance of each architecture quality attribute depends on the context and the stakeholders concerns for which the specific architecture is conceptualized.

Note As an illustration of Architecture quality attributes, either the ISO 9126 or the subsequent ISO 25010 product quality attributes can be considered. Figure below shows list of quality attributes and sub-attributes from ISO 25010.

Table Annex D-1: Quality attributes and sub-attributes from ISO 25010

Attribute	Sub-attribute	Attribute	Sub-attribute
Functional suitability	Functional completeness	Reliability	Maturity
	Functional correctness		Availability
	Functional appropriateness		Fault-tolerance
Performance efficiency	Time behaviour		Recoverability
	Resource utilization	Security	Confidentiality
	Capacity		Integrity
Compatibility	Co-existence		Non-repudiation
	Interoperability		Accountability
Usability	Appropriateness recognisability		Authenticity
	Learnability	Maintainability	Modularity
	Operability		Reusability
	User error protection		Analysability
	User interface aesthetics		Modifiability
	Accessibility		Testability

Annex E (informative) Architecture process-enabling resources

Examples of resources that can be used in performing the architecture processes are listed below.

- Architecture patterns: a pattern addresses a specific architecture problem, in a context, to provide a solution with a range of trade-offs
- Architecture styles: an idiom for organizing an architecture to achieve certain properties
- Model kinds: conventions for a type of modelling, to address specific types of concerns [ISO/IEC/IEEE 420101]
- Architecture description languages: any form of expression for use in architecture descriptions [ISO/IEC/IEEE 42010]
- Architecture viewpoints: work product establishing the conventions for the construction, interpretation and use of architecture views to frame specific system concerns [ISO/IEC/IEEE 42010]
- Architecture frameworks: conventions, principles and practices for the description of architectures established within a specific domain of application and/or community of stakeholders [ISO/IEC/IEEE 42010]
- Architecture methods: practice, technique, or procedure with rules to guide architecting
- Skills and knowledge associated to specific roles identified to perform architecture-related activities:
  - The skills required by each role
  - The depth of knowledge required to fulfil the role successfully
- Norms and Standards associated to the activities and the work products.
- Tools and languages sustaining the activities and allowing to formalise the work products and their related information.

NOTE Catalogues can be used to collect homogeneous sets of metadata, resources and related information. The repositories can be implemented with catalogues used as references for governance, management and usage.