INTERNATIONAL STANDARD

ISO/IEC/ IEEE 42010

First edition
2011-12-01

Systems and software engineering — Architecture description

Ingénierie des systèmes et des logiciels — Description de l'architecture

Reference number
ISO/IEC/IEEE 42010:2011(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York • NY 10016-5997, USA
E-mail stds.ipr@ieee.org
Web www.ieee.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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

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

The main task of ISO/IEC JTC 1 is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is called to the possibility that implementation of this standard may require the use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. ISO/IEEE is not responsible for identifying essential patents or patent claims for which a license may be required, for conducting inquiries into the legal validity or scope of patents or patent claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance or a Patent Statement and Licensing Declaration Form, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from ISO or the IEEE Standards Association.

ISO/IEC/IEEE 42010 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*, in cooperation with the Software and Systems Engineering Standards Committee of the Computer Society of the IEEE, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

This first edition of ISO/IEC/IEEE 42010 cancels and replaces ISO/IEC 42010:2007, which has been technically revised.

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.

Conceptualization of a system's architecture, as expressed in an architecture description, assists the understanding of the system's essence and key properties pertaining to its behaviour, composition and evolution, which in turn affect concerns such as the feasibility, utility and maintainability of the system.

Architecture descriptions are used by the parties that create, utilize and manage modern systems to improve communication and co-operation, enabling them to work in an integrated, coherent fashion. Architecture frameworks and architecture description languages are being created as assets that codify the conventions and common practices of architecting and the description of architectures within different communities and domains of application.

This International Standard addresses the creation, analysis and sustainment of architectures of systems through the use of architecture descriptions.

This International Standard provides a core ontology for the description of architectures. The provisions of this International Standard serve to enforce desired properties of architecture descriptions. This International Standard also specifies provisions that enforce desired properties of architecture frameworks and architecture description languages (ADLs), in order to usefully support the development and use of architecture descriptions. This International Standard provides a basis on which to compare and integrate architecture frameworks and ADLs by providing a common ontology for specifying their contents.

This International Standard can be used to establish a coherent practice for developing architecture descriptions, architecture frameworks and architecture description languages within the context of a life cycle and its processes (not defined by this International Standard). This International Standard can further be used to assess conformance of an architecture description, of an architecture framework, of an architecture description language, or of an architecture viewpoint to its provisions.

Users of this International Standard are advised to consult Clause 4 to gain appreciation of the provided ontology, its concepts and principles.

Systems and software engineering — Architecture description

1 Scope

This International Standard specifies the manner in which architecture descriptions of systems are organized and expressed.

This International Standard specifies architecture viewpoints, architecture frameworks and architecture description languages for use in architecture descriptions.

This International Standard also provides motivations for terms and concepts used; presents guidance on specifying architecture viewpoints; and demonstrates the use of this International Standard with other standards.

2 Conformance

The requirements in this International Standard are contained in Clauses 5, 6 and 7. There are four situations in which claims of conformance with the provisions of this International Standard can be made.

— When conformance is claimed for an architecture description, the claim shall demonstrate that the architecture description meets the requirements listed in Clause 5.

— When conformance is claimed for an architecture viewpoint, the claim shall demonstrate that the architecture viewpoint meets the requirements listed in Clause 7.

— When conformance is claimed for an architecture framework, the claim shall demonstrate that the architecture framework meets the requirements listed in 6.1.

— When conformance is claimed for an architecture description language, the claim shall demonstrate that the architecture description language meets the requirements listed in 6.3.

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”. In the event of a conflict between normative figures and text, the text takes precedence. Please report any apparent conflicts.

NOTE This International Standard is designed such that “tailoring” is neither required nor permitted for its use when claims of conformance are made.

3 Terms and definitions

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

3.1
architecting
process of conceiving, defining, expressing, documenting, communicating, certifying proper implementation of, maintaining and improving an architecture throughout a system’s life cycle

NOTE 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") [ISO/IEC 12207, ISO/IEC 15288].

3.2
architecture
⟨system⟩ fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution

3.3
architecture description
AD
work product used to express an architecture

3.4
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.

3.5
architecture view
work product expressing the architecture of a system from the perspective of specific system concerns

3.6
architecture viewpoint
work product establishing the conventions for the construction, interpretation and use of architecture views to frame specific system concerns

3.7
concern
⟨system⟩ interest in a system relevant to one or more of its stakeholders

NOTE A concern pertains to any influence on a system in its environment, including developmental, technological, business, operational, organizational, political, economic, legal, regulatory, ecological and social influences.

3.8
environment
⟨system⟩ context determining the setting and circumstances of all influences upon a system

NOTE The environment of a system includes developmental, technological, business, operational, organizational, political, economic, legal, regulatory, ecological and social influences.

3.9
model kind
conventions for a type of modelling

NOTE Examples of model kinds include data flow diagrams, class diagrams, Petri nets, balance sheets, organization charts and state transition models.

3.10
stakeholder
⟨system⟩ individual, team, organization, or classes thereof, having an interest in a system

4 Conceptual foundations

4.1 Introduction

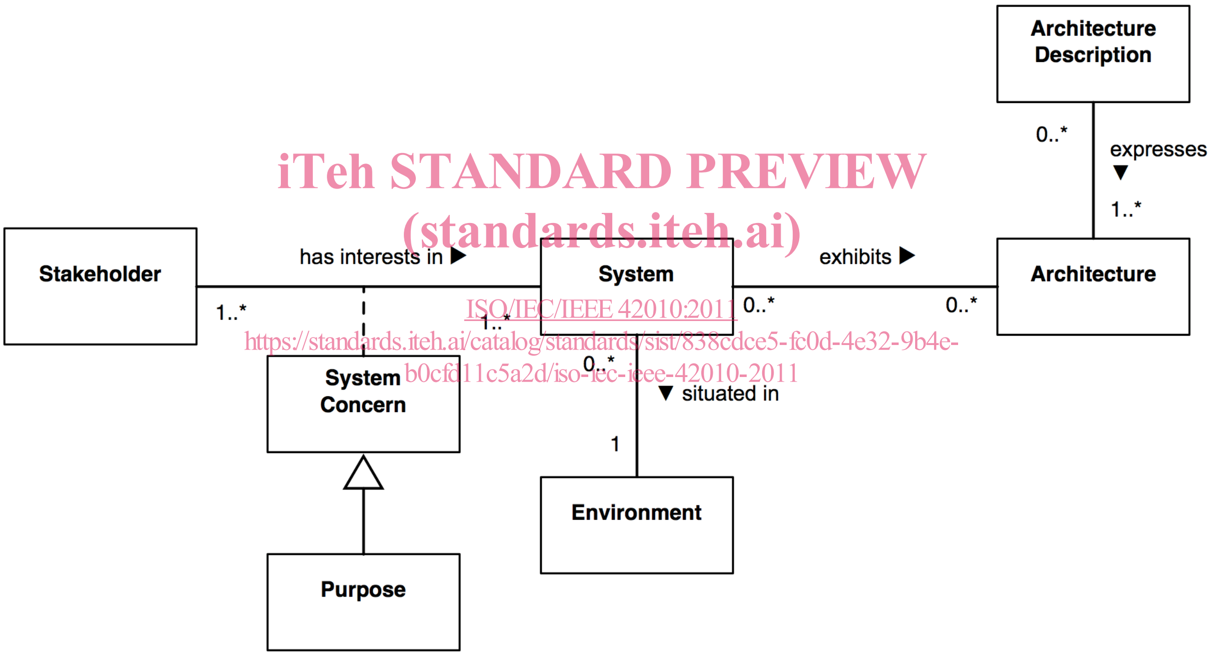
This clause introduces the conceptual foundations of architecture description comprising a conceptual model of architecture description (see 4.2); the role of architecting in the life cycle (see 4.3); uses of architecture descriptions (see 4.4); and architecture frameworks and architecture description languages (see 4.5). The concepts introduced in this clause are used in Clauses 5 through 7 to express requirements.

NOTE Annex A provides further discussion of the terms and concepts used in this International Standard and presents examples of their use.

4.2 Conceptual model of architecture description

4.2.1 Context of architecture description

Figure 1 depicts key concepts pertaining to systems and their architectures as a context for understanding the practice of architecture description.



NOTE The figure uses the conventions for class diagrams defined in [ISO/IEC 19501].

Figure 1 — Context of architecture description

The term *system* is used in this International Standard to refer to entities whose architectures are of interest. The term is intended to encompass, but is not limited to, entities within the following domains:

— **systems** as described in [ISO/IEC 15288]: “systems that are man-made and may be configured with one or more of the following: hardware, software, data, humans, processes (e.g., processes for providing service to users), procedures (e.g. operator instructions), facilities, materials and naturally occurring entities”;

— **software products and services** as described in [ISO/IEC 12207];

— **software-intensive systems** as described in [IEEE Std 1471:2000]: “any system where software contributes essential influences to the design, construction, deployment, and evolution of the system as a whole” to encompass “individual applications, systems in the traditional sense, subsystems, systems of systems, product lines, product families, whole enterprises, and other aggregations of interest”.

This International Standard takes no position on what constitutes a system within those domains—or elsewhere. The nature of systems is not defined by this International Standard.

This International Standard is intended for use in the domains of systems listed above; however, nothing herein precludes its use for architecture descriptions of entities of interest outside of those domains (for example, natural systems and conceptual systems).

The stakeholders of a system are parties with interests in that system. Stakeholders’ interests are expressed as concerns (see 4.2.3). Stakeholders ascribe various *purposes* to a system. Purposes are one kind of concern.

NOTE 1 The term *purpose* as used in this International Standard derives from its use in ISO/IEC 15288:2008, 4.31: a system is a combination of interacting elements organized to achieve one or more stated purposes.

A system is situated in an environment. The environment determines the totality of influences upon the system throughout its life cycle, including its interactions with that environment. The environment of a system can contain other systems.

NOTE 2 In this International Standard, the environment of a system is bounded by and understood through the identification and analysis of the system’s stakeholders and their concerns (see 4.2.3).

The architecture of a system constitutes what is essential about that system considered in relation to its environment. There is no single characterization of what is essential or fundamental to a system; that characterization could pertain to any or all of:

— system constituents or elements;

— how system elements are arranged or interrelated;

— principles of the system’s organization or design; and

— principles governing the evolution of the system over its life cycle.

Architecture descriptions are used to express architectures for systems of interest (see 4.2.2).

NOTE 3 The same system could be understood through several distinct architectures (for example, when considered in different environments). An architecture could be expressed through several distinct architecture descriptions (for example when different architecture frameworks are employed). The same architecture could characterise more than one system (for example a family of systems sharing a common architecture)

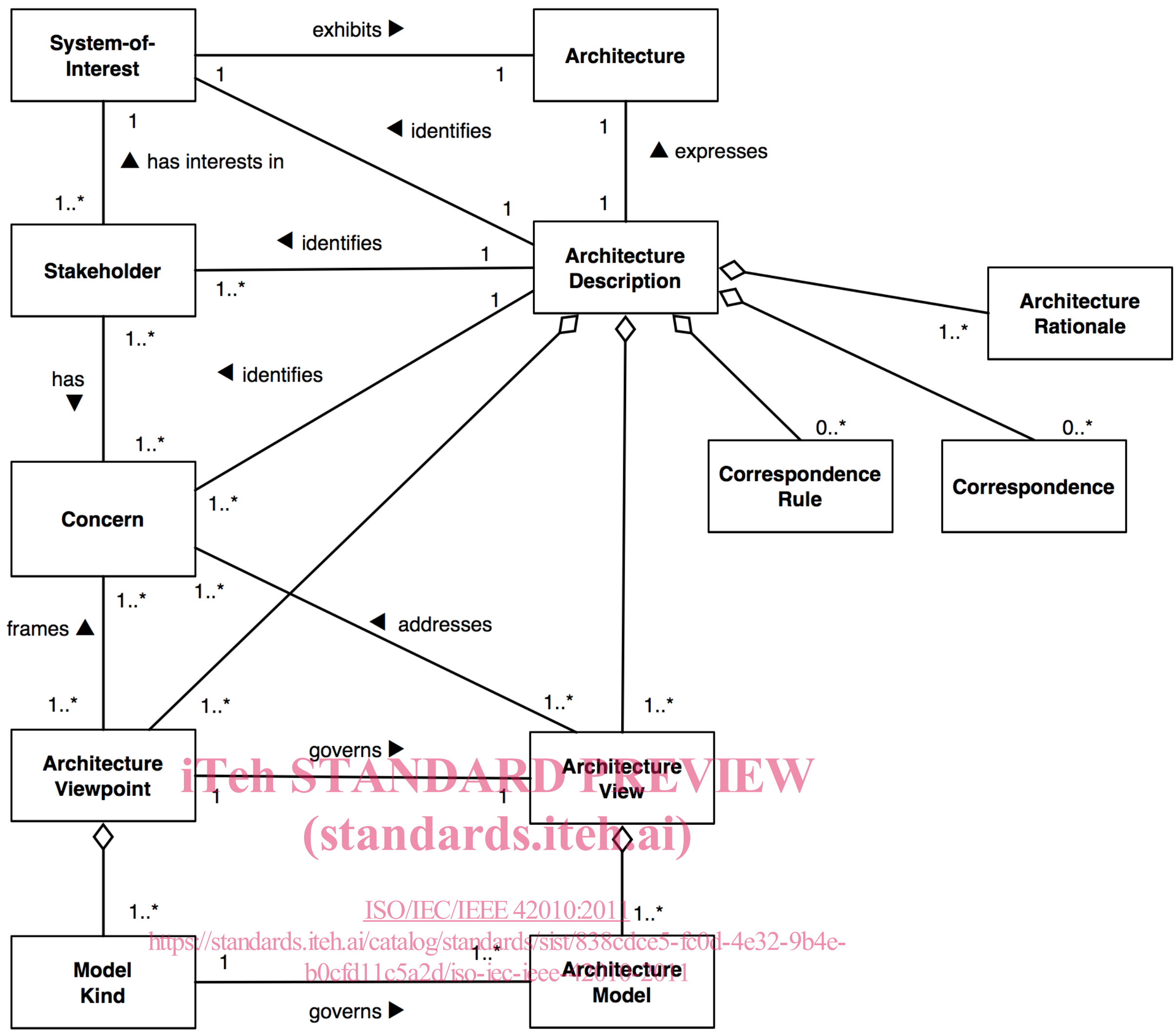
4.2.2 Architectures and architecture descriptions

Architecture descriptions are work products of systems and software architecting.

Figure 2 depicts concepts pertaining to the practice of architecture description *when applying this International Standard to produce one architecture description expressing one architecture for one system-of-interest*.

In this International Standard, the term *system-of-interest* (or simply, *system*) refers to the system whose architecture is under consideration in the preparation of an architecture description.

The figures and text in the remainder of 4.2 constitute a *conceptual model* of architecture description.



NOTE 1 The figure uses the conventions for class diagrams defined in [ISO/IEC 19501].

NOTE 2 Figure 3 provides additional details on correspondences and correspondence rules. Figure 4 provides additional details on architecture rationale.

Figure 2 — Conceptual model of an architecture description

An architecture description expresses an architecture of a system-of-interest.

This International Standard distinguishes an *architecture of a system* from an *architecture description*. Architecture descriptions, not architectures, are the subject of this International Standard. Whereas an architecture description is a work product, an architecture is abstract, consisting of concepts and properties. This International Standard specifies requirements on architecture descriptions. There are no requirements in this International Standard pertaining to architectures, or to systems or to their environments.

This International Standard does not specify any format or media for recording architecture descriptions. It is intended to be usable for a range of approaches to architecture description including document-centric, model-based, and repository-based techniques.

This International Standard does not prescribe the process or method used to produce architecture descriptions. This International Standard does not assume or prescribe specific architecting methods, models, notations or techniques used to produce architecture descriptions.

4.2.3 Stakeholders and concerns

Stakeholders of a system have concerns with respect to the system-of-interest considered in relation to its environment. A concern could be held by one or more stakeholders. Concerns arise throughout the life cycle from system needs and requirements, from design choices and from implementation and operating considerations. A concern could be manifest in many forms, such as in relation to one or more stakeholder needs, goals, expectations, responsibilities, requirements, design constraints, assumptions, dependencies, quality attributes, architecture decisions, risks or other issues pertaining to the system.

EXAMPLES The following are concerns in the terms of this International Standard: functionality, feasibility, usage, system purposes, system features, system properties, known limitations, structure, behavior, performance, resource utilization, reliability, security, information assurance, complexity, evolvability, openness, concurrency, autonomy, cost, schedule, quality of service, flexibility, agility, modifiability, modularity, control, inter-process communication, deadlock, state change, subsystem integration, data accessibility, privacy, compliance to regulation, assurance, business goals and strategies, customer experience, maintainability, affordability and disposability. The *distribution transparencies* described in the Reference Model of Open Distributed Processing [ISO/IEC 10746-1] are concerns in the terms of this International Standard. *Software properties* as described in SQUARE [ISO/IEC 25010:2011, 4.2] name concerns in the terms of this International Standard.

4.2.4 Architecture views and viewpoints

An architecture description includes one or more architecture views. An architecture view (or simply, *view*) addresses one or more of the concerns held by the system's stakeholders.

An architecture view expresses the architecture of the system-of-interest in accordance with an architecture viewpoint (or simply, *viewpoint*). There are two aspects to a viewpoint: the concerns it frames for stakeholders and the conventions it establishes on views.

An architecture viewpoint **frames**[1] one or more concerns. A concern can be framed by more than one viewpoint.

A view is **governed** by its viewpoint: the viewpoint establishes the conventions for constructing, interpreting and analyzing the view to address concerns framed by that viewpoint. Viewpoint conventions can include languages, notations, model kinds, design rules, and/or modelling methods, analysis techniques and other operations on views.

Figure 2 depicts the relations between views and viewpoints within an architecture description.

NOTE 1 This International Standard does not use phrases such as “business architecture”, “physical architecture”, and “technical architecture”. In the terms of this International Standard, the architecture of a system is a holistic conception of that system's fundamental properties, best understood via multiple views of that architecture. Therefore, approximate equivalents of the above phrases are “business view”, “physical view”, and “technical view”, respectively.

NOTE 2 Clause 7 specifies requirements on architecture viewpoints. Annex B provides guidance on specifying viewpoints.

4.2.5 Architecture models

An architecture view is composed of one or more architecture models. An architecture model uses modelling conventions appropriate to the concerns to be addressed. These conventions are specified by the *model kind* governing that model. Within an architecture description, an architecture model can be a part of more than one architecture view.

Figure 2 depicts the use of architecture models and model kinds within an architecture description.

NOTE This International Standard uses the term *model kind* rather than “architecture model kind” to emphasize that model kinds need not be useful exclusively in architecture descriptions.

---

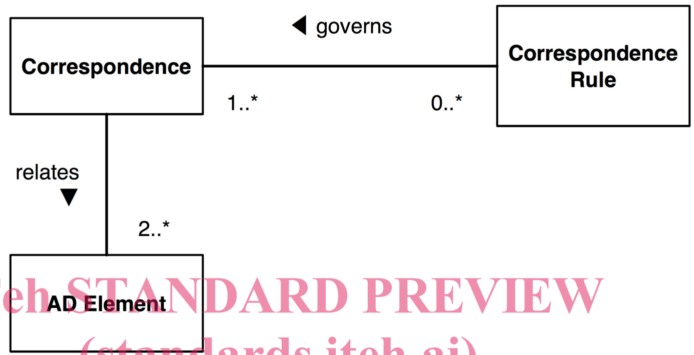
[1] In this International Standard, the verb *frame* is used in its ordinary language sense: to formulate or construct in a particular style or language; to enclose in or as if in a frame; to surround so as to create a sharp or attractive image.

4.2.6 AD elements and correspondences

An *AD element* is any construct in an architecture description. AD elements are the most primitive constructs discussed in this International Standard. Every stakeholder, concern, architecture viewpoint, architecture view, model kind, architecture model, architecture decision and rationale (see 4.2.7) is considered an AD element. When viewpoints and model kinds are defined and their models are populated, additional AD elements are introduced.

A *correspondence* defines a relation between AD elements. Correspondences are used to express architecture relations of interest within an architecture description (or between architecture descriptions). Correspondences can be governed by *correspondence rules*. Correspondence rules are used to enforce relations within an architecture description (or between architecture descriptions).

Figure 3 depicts the concepts of AD elements and correspondences.



NOTE The figure uses the conventions for class diagrams defined in [ISO/IEC 19501].

Figure 3 — Conceptual model of AD elements and correspondences

EXAMPLES Correspondences and correspondence rules are used to express and enforce architecture relations such as composition, refinement, consistency, traceability, dependency, constraint and obligation.

NOTE Requirements on using correspondences and correspondence rules are specified in 5.7. Examples of their use are given in A.6.

4.2.7 Architecture decisions and rationale

Architecture rationale records explanation, justification or reasoning about architecture decisions that have been made. The rationale for a decision can include the basis for a decision, alternatives and trade-offs considered, potential consequences of the decision and citations to sources of additional information.

Decisions pertain to system concerns; however, there is often no simple mapping between the two. A decision can affect the architecture in several ways. These can be reflected in the architecture description as follows:

— requiring the existence of AD elements;

— changing the properties of AD elements;

— triggering trade-off analyses in which some AD elements, including other decisions and concerns, are revised;

— raising new concerns.

Figure 4 depicts concepts pertaining to architecture decisions and rationale.