ISO/IEC 30141

Edition 2.0 2024-08

INTERNATIONAL STANDARD

colour inside

Internet of things (IoT) – Reference architecture

ISO/IEC 30141:2024-08(en)

IEC Secretariat
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
info@iec.ch
www.iec.ch

ISO/IEC 30141

Edition 2.0 2024-08

INTERNATIONAL STANDARD

colour inside

Internet of things (IoT) – Reference architecture

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 35.020

ISBN 978-2-8322-9193-1

CONTENTS

INTERNET OF THINGS (IoT) – REFERENCE ARCHITECTURE

FOREWORD

1) 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.
2) The formal decisions or agreements of IEC and ISO on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC and ISO National bodies.
3) IEC and ISO documents have the form of recommendations for international use and are accepted by IEC and ISO National bodies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC and ISO documents is accurate, IEC and ISO cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC and ISO National bodies undertake to apply IEC and ISO documents transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC and ISO document and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC and ISO do not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC and ISO marks of conformity. IEC and ISO are not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this document.
7) No liability shall attach to IEC and ISO or their directors, employees, servants or agents including individual experts and members of its technical committees and IEC and ISO National bodies for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this ISO/IEC document or any other IEC and ISO documents.
8) Attention is drawn to the Normative references cited in this document. Use of the referenced publications is indispensable for the correct application of this document.
9) IEC and ISO draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). IEC and ISO take no position concerning the evidence, validity, or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, IEC and ISO had not received notice of (a) patent(s), which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at https://patents.iec.ch and www.iso.org/patents. IEC and ISO shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 30141 has been prepared by subcommittee 41: Internet of Things and Digital Twin, of ISO/IEC joint technical committee 1: Information technology. It is an International Standard.

This second edition cancels and replaces the first edition published in 2018. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

a) conformance with ISO/IEC/IEEE 42010:2022;
b) improved usability;
c) implementation pattern support.

The text of this International Standard is based on the following documents:

Draft	Report on voting
JTC1-SC41/417/FDIS	JTC1-SC41/431/RVD

Full information on the voting for its approval can be found in the report on voting indicated in the above table.

The language used for the development of this International Standard is English.

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1, and the ISO/IEC Directives, JTC 1 Supplement available at www.iec.ch/members_experts/refdocs and www.iso.org/directives.

IMPORTANT – The "colour inside" logo on the cover page of this document indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

0.1 General

This document is the second edition of the Internet of Things reference architecture (IoT RA). This document is in conformance with ISO/IEC/IEEE 42010:2022 requirements on architecture descriptions that are described in Annex B except for aspects, perspectives, decisions and rationale.

The IoT RA addresses systems that:

- use technology for sensing and operating on physical world entities;
- have components that interact through a digital network.

The IoT RA deals with related issues, like trustworthiness functions regarding both the physical and digital worlds.

This document can be used as a generic normative part of IoT domain-specific reference architectures (DSRAs). IoT DSRAs are in conformance with ISO/IEC/IEEE 42010:2022 requirements on architecture descriptions and enable consistency and alignment with other reference architectures within ISO/IEC JTC 1.

0.2 About Internet of Things (IoT)

IoT has broad implications in industry and society today and is likely to continue to have an impact on many aspects of our lives for many years to come. Various IoT applications and services have adopted IoT techniques to provide capabilities that were not possible earlier. IoT is one of the most dynamic areas of information and communication technologies.

Fundamental to IoT are devices that interact with the physical world. Sensors collect the information about the physical world, while actuators can act upon the physical world. These field devices are connected to the digital world through network connections. Both sensors and actuators can be in many forms such as thermometers, accelerometers, video cameras, microphones, relays, heaters or industrial equipment for manufacturing or process control.

IoT is the base for new business models or offerings and new working methods in industry and in the public sector. IoT is an essential enabler for other computing areas such as digital twins, artificial intelligence, cloud computing, big data, data analysis and more. Many application areas called "smart xxx" such as smart grid, smart cities, and even smart cars use IoT as an important technology capability.

IoT can be combined with other technologies to address complex requirements. For example, IoT can leverage cloud computing, including private cloud, public cloud, hybrid cloud, and multi-cloud, for resource provisioning and management. IoT can benefit from machine learning and big data for the analysis of sensor data to enable rapid decisions for improved control and efficiency. IoT with distributed ledger technology can ensure traceability in applications. IoT can take advantage of edge computing to distribute computing resources near the convergence of information technology and operational technology, where they are needed most. The IoT area continues to grow rapidly, and new IoT application areas continue to be found and invented. This document can serve these new technology and application areas.

0.3 IoT sources of information

For a given application field and purpose, the many IoT standards, guidelines, and initiatives in existence today work well on their own and are used by various IoT stakeholders. As a result, heterogeneity is a prominent aspect of IoT. However, support for the combination and interaction of these heterogeneous resources to enable interoperability and convergence between IoT standards and guidelines is necessary.

Stakeholder decisions about both a foundation for long-term investments and durable protection of current cornerstones is more difficult because of uncertainty about resource compatibility.

This document serves as a foundation for creating interoperability and alignment between IoT initiatives. The aim of this document is to bring different views together.

0.4 General principles of a reference architecture

This document is positioned as a reference architecture for IoT systems. It utilizes the terms, definitions, and relationships for best practices in architecture descriptions as outlined in ISO/IEC/IEEE 42010:2022 to:

- establish vocabulary, principles, guidance; and
- provide a description of IoT principles, capabilities, and interactions with the physical and digital worlds.

One of the primary purposes of the IoT RA is to support architects that want to design architectures or reference architectures for IoT systems. Normative parts of the IoT RA can then be included in an architecture closer to the realization of IoT systems.

Figure 1 shows how this document has been specified and how it will be used.

– This document conforms to ISO/IEC/IEEE 42010:2022 requirements for architecture descriptions [1][1] and uses guidelines from the "Best practices and guidelines for RA standards" standing document [2].

– Users of this document apply it to specify an IoT architecture that guides the implementation of an IoT system.

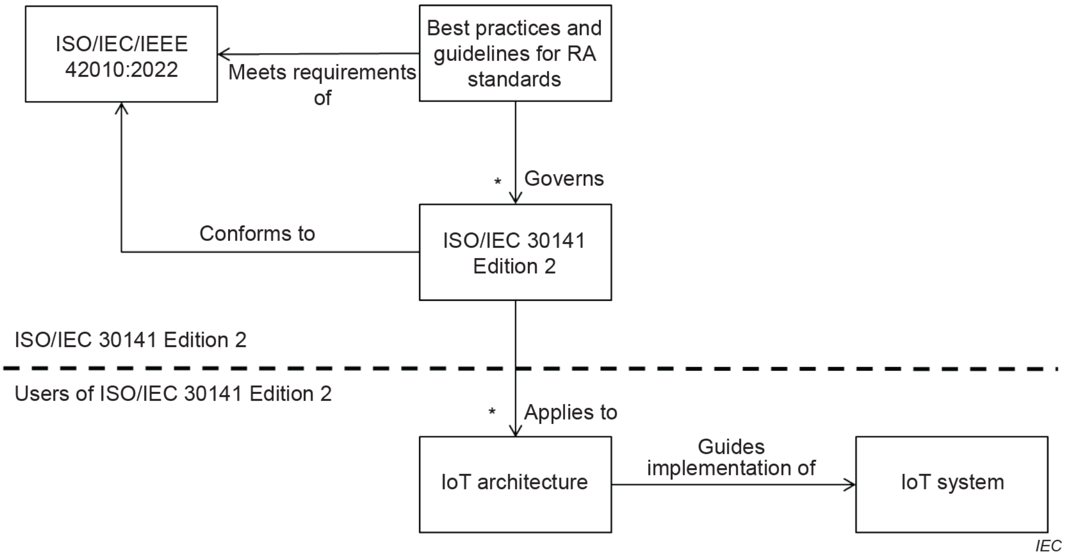


Figure 1 – Using the IoT RA standard

[1] Numbers in square brackets refer to the Bibliography.

INTERNET OF THINGS (IoT) – REFERENCE ARCHITECTURE

1 Scope

This document specifies an Internet of Things (IoT) reference architecture (IoT RA). The IoT RA is a generalization of existing practice including the distinguishing characteristics of IoT systems and other fundamental characteristics exhibited by IoT systems. The IoT RA addresses stakeholder concerns related to the business value of IoT systems. The IoT RA also addresses the interactions between the IoT system, the users, and the physical environment. Implementation of IoT systems is also addressed in this document. Among the characteristics specified in the IoT RA are abstract functions within IoT systems and a variety of structures that are used to construct IoT systems.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 20924, *Internet of Things (IoT) and digital twin – Vocabulary*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 20924 apply.

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

- ISO Online browsing platform: available at https://www.iso.org/obp
- IEC Electropedia: available at https://www.electropedia.org

4 Abbreviated terms

AI	artificial intelligence
API	application programming interface
ASD	application and service domain
BSS	business support systems
CM	conceptual model
CPS	cyber physical system
CPO	chief privacy officer
DPO	data protection officer
DSC	dynamic service composition
DSRA	domain specific reference architecture
ETL	extract, transform, load
HMI	human machine interface
HTTP	Hypertext Transfer Protocol
ICT	information and communication technologies

IoT	Internet of Things
IT	information technology
LAN	local area network
LoB	line of business
OMD	operation and management domain
OSS	operational support systems
OT	operational technology
PED	physical entity domain
PII	personally identifiable information
RA	reference architecture
RAID	resource access and interchange domain
SCD	sensing and controlling domain

5 IoT RA context

5.1 Overview

This document specifies a reference architecture for the Internet of Things using the ISO/IEC/IEEE 42010:2022 requirements for architecture descriptions. While this document is intended to be used in many situations and by many different stakeholders, the focus is on the needs of architecture developers of an IoT system description, their perspectives, concerns, and the architecture views they most likely encounter. In particular, it focuses on those involved in creating a specific IoT reference or solution architecture that depends on IoT.

This document is interesting and educational for anyone with interest in IoT. However, the document is of special interest for those people looking to:

- harmonize standards or reference architectures of other IT and OT domains with this generic IoT RA;
- develop either IoT software, or IoT hardware, or both;
- provide services involving IoT;
- procure or implement IoT systems;
- integrate IoT with other IT technology.

For all the interests mentioned above, applying parts of this document to develop or implement IoT domain-specific architecture (DSRA) is a useful approach. These descriptions can be either standalone or integrated with other IoT related architecture descriptions.

Further, depending on the interest, different sections of the document have more relevancy. While some parts provide an overview, some parts provide more information regarding different aspects of IoT.

This document can also be used as normative reference in a DSRA. The document can then serve as a base for the IoT part of that DSRA, leaving application concerns, use cases and so forth to the DSRA. The DSRA can specify how the IoT RA is intended to be used for the specific domain. A DSRA can also specify conformance to parts of this document.

This document conforms to ISO/IEC/IEEE 42010:2022 requirements on architecture descriptions [1], which is described in Annex B. This ensures that the IoT RA aligns with other RAs that conform to ISO/IEC/IEEE 42010:2022 requirements on architecture descriptions. The IoT RA is intended to be used with one or more architecture description frameworks (in accordance with ISO/IEC/IEEE 42010:2022) to create an architecture description for a system of interest.

ISO/IEC/IEEE 42010:2022 provides terms, definitions, and relationships for best practices in architecture descriptions of the architecture of a system of interest. Consequently, all conforming architecture descriptions can be interpreted in a consistent way. Conformance to ISO/IEC/IEEE 42010:2022 requirements for architecture descriptions makes this document a vital part of a cohesive family of standards, including those standards dealing with interoperability.

5.2 Stakeholders and concerns

Table 1 shows a list of viewpoints, stakeholders, and concerns.

Table 1 – List of viewpoints, stakeholders, and concerns

Viewpoint	Stakeholders	Concerns
Foundational IoT viewpoint	– Architect – Project manager – Programme manager – Standards expert – People concerned with the fundamentals of IoT – Domain experts – Business manager – System owner	What is IoT? What are the essential characteristics of IoT systems? What is new (and different) about the concept of IoT? Is a given system or component an IoT system or component? What are the implications of the concept of IoT?
Business viewpoint	– Business manager – System owner – Architect	How to leverage the various capabilities of an IoT system to provide value for a business? How to use IoT for innovative new business models? How do characteristics of an IoT system influence business and system owner?
Usage viewpoint	– System architect – Project manager	How do users (both human and digital) interact with the IoT system? How does the IoT system interact with the physical entity of interest?
Functional viewpoint	– Architect – Project manager – Programme manager – IoT standards expert – Business manager – System owner	What are the types of abstract functions that need to be implemented in an IoT system?
Trustworthiness viewpoint	– System architect – Security engineer – Security manager – Privacy manager – Project manager – Business manager	How to design an IoT system with a level of confidence that meets trustworthiness goals. How to assure the implemented system meets design goals.
Construction viewpoint	– System architect – Project manager – System designer	What types of design are useful when creating an IoT system to meet a given set of requirements? What types of design are useful when creating an IoT component to meet a given set of requirements?

6 IoT RA viewpoints and views

6.1 Overview

The essence of an architecture description lies in the architecture views that it provides. Each view is governed by a corresponding viewpoint. The developers and professional practitioners who are charged with elaboration of the architecture description can adopt the architecture views provided in this document. The different architecture views result from modelling and narrative descriptive activities of the architecting effort.

An architecture viewpoint states relevant information focusing on a particular topic. This information is a collection of concerns, often expressed in use cases, stakeholders provide about the constructed artefacts that they believe are important to the topic. Viewpoints specify the architecture view or views it governs. The architecture viewpoint also identifies modelling or narrative paradigms for generating the architecture view or views.

This document specifies six architecture views and their governing viewpoints:

- an IoT foundational viewpoint in 6.2;
- a business viewpoint in 6.3;
- a usage viewpoint in 6.4;
- a functional viewpoint in 6.5;
- a trustworthiness viewpoint in 6.6;
- a construction viewpoint in 6.7.

Subclause 6.2 presents the foundational IoT viewpoint and the corresponding view which frames the concerns related to the essential characteristics of IoT. The viewpoint addresses concerns related to the fundamental aspects of IoT by specifying the foundational building blocks of IoT enabled systems as architecture views. These views represent the different concepts related to IoT (for example: principles, devices, and connectivity).

Subclause 6.3 presents the business viewpoint and the corresponding view.

Subclause 6.4 presents the usage viewpoint and the corresponding view.

Subclause 6.5 presents the functional viewpoint and the corresponding functional views that describe the IoT reference architecture from the perspective of key functions. The viewpoint addresses concerns related to the fundamental functional aspects of IoT by specifying the foundational building blocks of IoT enabled systems as architecture views. These views represent the functional capabilities of IoT related data, management, communication, interfaces, etc.

Subclause 6.6 presents the trustworthiness viewpoint and provides means for implementing a trust model to consider when creating an IoT product or solution.

Subclause 6.7 presents the construction viewpoint and the corresponding construction views that describe the IoT reference architecture from the perspective of implementation architectures. The viewpoint addresses concerns related to the fundamental implementation aspects of IoT by specifying the different patterns as architecture views. Examples of patterns described in this document include the IoT component capability, the RAMI4.0, the IoT user, the IoT enterprise system, the enterprise networking, and the IoT enterprise usage patterns.

6.2 Foundational IoT viewpoint and views

6.2.1 Foundational IoT viewpoint

The foundational IoT viewpoint is essential to the IoT domain. It frames multiple concerns of several stakeholders, described below. The foundational IoT viewpoint is described in Table 2.

Table 2 – Foundational IoT viewpoint

Viewpoint name		Foundational IoT
Overview		Framing of the concerns related to the essential characteristics of IoT
Known typical stakeholders		– Architect – Project manager – Programme manager – Standards expert – People concerned with the fundamentals of IoT – Domain experts – Business managers – System owners
Concerns		– What is IoT? – What are the essential characteristics of IoT systems? – What is new (and different) about the concept of IoT? – Is a given system or component an IoT system or component? – What are the implications of the concept of IoT?
Viewpoint specification	Model kinds	Interoperability model Component model
	Legends	The foundational IoT view provides text defining the essential characteristics of IoT, text describing the concepts and relationships between an IoT component, IoT system, and IoT environment.
	View methods	NA
	Correspondence methods	NA
	References	NA

6.2.2 Foundational IoT view

6.2.2.1 Description of IoT systems

ISO/IEC 20924 [3] defines IoT as an "infrastructure of interconnected entities, people, systems and information resources together with services which processes and reacts to information from the physical world and virtual world".

In practical terms, this IoT definition can be distilled down to two essential characteristics of IoT systems.

a) An IoT system is composed of components connected through a many-to-many digital network. The network capabilities can be based on different options such as using a many-to-many relationship or others and using approaches such as TCP/IP or alternatives.

b) At least one of those system components interacts with the physical world through either sensing or actuating.

The interaction with the physical world includes:

– Sensing: a sensing capability (provided by a sensor) offers ability to provide observations of an aspect of the physical world as measurement data. Information from sensor observations can be provided to other IoT components through the network interface of the component for processing and storage.

  EXAMPLE 1 Temperature sensing (temperature measurement capability), computerized tomography (CT) scans (radiographic imaging), spatial sensing (accelerometers, gyroscopes), optical sensing, and audio sensing.

– Actuating: an actuating capability, provided by an actuator, offers the ability to change the physical world. Such change is based on information that is given as input to the component.

  EXAMPLE 2 Heating coils (heating capability), electric shock delivery (cardiac pacing), electronic door locks (lock/unlock capability), unmanned aerial vehicle operation (remote control), servo motors (motion/movement capability), and robotic arms (complex motion/movement capability).

6.2.2.2 IoT system concepts

6.2.2.2.1 General

IoT can be broken down into three important concepts (Figure 2):

– the IoT environment (containing all the components, systems, and related infrastructure);

– the IoT system that provides benefit to the stakeholders;

– the IoT components that interact together to form the IoT system.

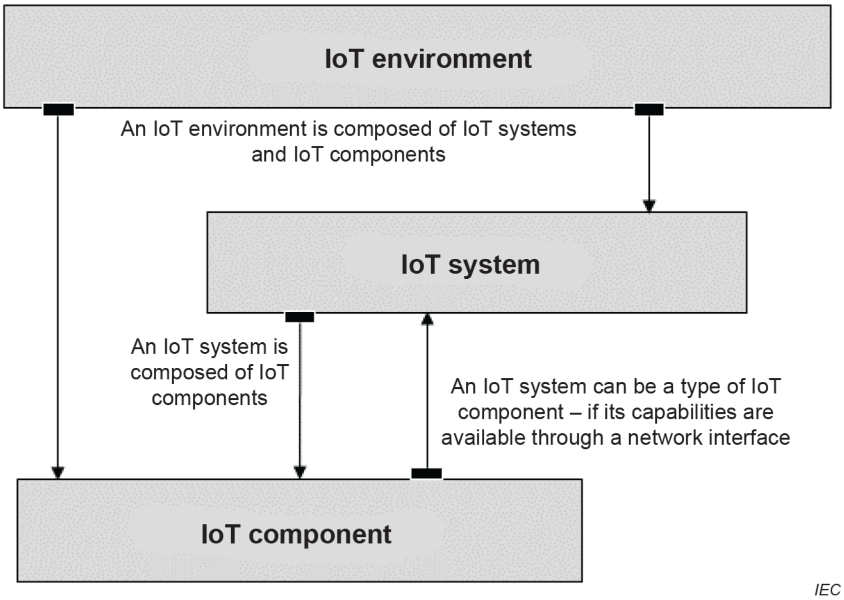

IoT environment
An IoT environment is composed of IoT systems and IoT components
IoT system
An IoT system is composed of IoT components
An IoT system can be a type of IoT component – if its capabilities are available through a network interface
IoT component
IEC

Figure 2 – Relationship between IoT component, IoT system and IoT environment

6.2.2.2.2 IoT components

IoT components are the basic building blocks of IoT systems. IoT components interact with other IoT components to form a system and achieve one or more goals. Each IoT component provides some function that is necessary within the system so it can achieve its goal or goals.

All IoT components have at least one network interface that provides the ability to participate in a many-to-many network. Nevertheless, a given IoT component does not need to communicate with more than one other IoT component in a given system (e.g. assigning and limiting communication between two static IPs). Most IoT components also have an application interface that provides the capacity for application-level interactions between IoT components.

Each IoT component offers one or more IoT capabilities for use by other IoT components. Network interfaces and sensing are two examples of IoT capabilities.

6.2.2.2.3 IoT systems

A system, which is a combination of interacting elements organized to achieve one or more stated purposes, is considered an IoT system when it is composed of networked IoT components. The IoT system also interacts with a physical entity of interest through a sensor or an actuator within the IoT components. IoT systems differ from conventional IT systems in their ability to directly interact with the physical world.

IoT systems range from the very simple, such as an Internet-enabled thermometer, to the extremely complex, such as a city management system, and everything in between. IoT components can be assembled into many different systems. Also, a single IoT component can have the ability to be part of more than one system at a time. An IoT system can also act as an IoT component within another IoT system, if it has a network interface that allows it to be used that way. An IoT system is not necessarily constructed in a particular manner.

The quality of "goodness" is how well requirements of stakeholders are met. Many of the characteristics of an IoT system have lower or higher importance depending on the specific use cases and therefore indirectly stakeholder concerns.

An IoT system is not required to meet specific security, privacy, reliability, cost, or functional requirements to be considered an IoT system. Even if these criteria are not met it does not stop being an IoT system. However, if the IoT system cannot meet the use case requirements, it might not be a "good" IoT system for a specific use case. In other words, "goodness" is purely in the eyes of the user.

6.2.2.2.4 IoT environments

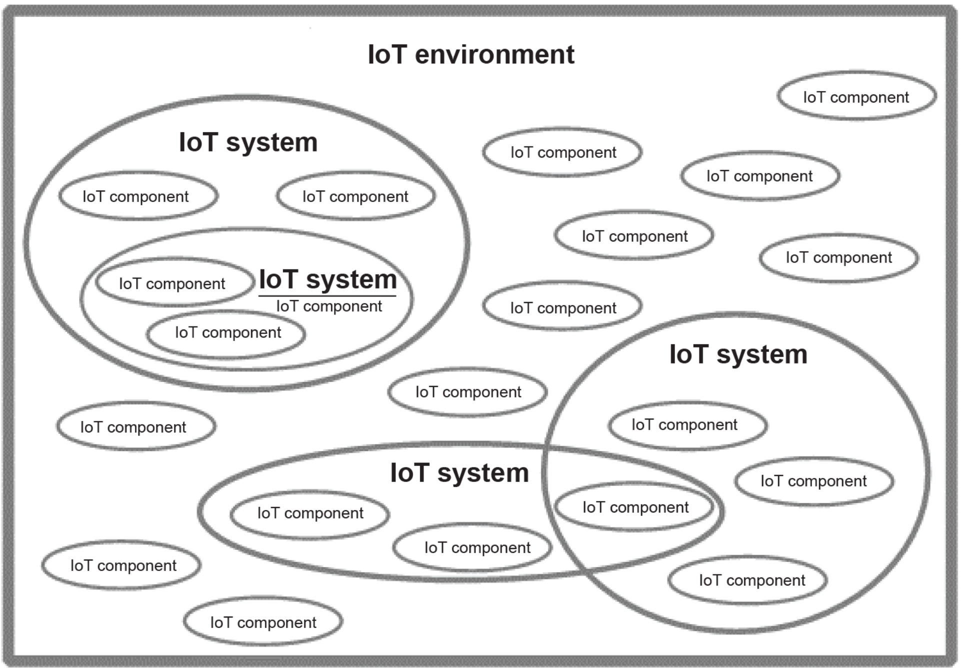
A digital IoT environment includes:

- the set of IoT components available to be composed into IoT systems;
- the networks connecting the components;
- any associated services that provide the mechanisms for discovery, composition, and orchestration.

Although an IoT environment contains the IoT components that can be used to create IoT systems, it does not necessarily contain any functioning IoT systems. This can be the case if no IoT components have been instructed to interact as a system. The opposite is also true. An IoT environment can be used to create non-IoT systems (conventional IT systems) by excluding IoT components that have sensing or actuating capabilities. In its current state, the Internet can be considered an IoT environment.

For system owners and business managers, it is often relevant to consider the IoT environment as a part of the overall IT environment of an organization.

IoT systems can be components of other IoT systems. IoT systems can also be components of other IT systems. IT systems can be (and often are) components of IoT systems.



IEC

Figure 3 – Example of IoT environment

Figure 3 shows an example of IoT environment with systems, components, and system of systems.

IoT components that are not part of a specified IoT system are nonetheless interconnected in the IoT environment and can become part of an IoT system.

6.2.2.3 Emergent characteristics

6.2.2.3.1 General

The characteristics inherent to IoT are described in 6.2.2.3.2 to 6.2.2.3.10.

NOTE Other characteristics that can be useful to build IoT systems in particular contexts are listed in Annex C.

6.2.2.3.2 Composability

Composability is the ability to combine discrete IoT components into an IoT system to achieve a set of goals and objectives. Composability and interoperability might be important to avoid lock in effects to one or a few IoT suppliers. Especially in the fast-growing IoT area, this might be even more important to meet expected delivery agreements with customers. When the customer demands exceed production capacities, an organization might need to have back up plans for how to upgrade or replace failing equipment.

6.2.2.3.3 Functional and management capability separation

Separation of functional and management capabilities means that the functional interfaces and capabilities of an IoT component, such as an IoT device, are cleanly separated from the management interfaces and capabilities of that component.

It is often wanted to qualify in an early stage the different kinds of users that need to interact with different parts of the IoT system and how. Such qualification makes the separation need more precise and specific. Since the IoT system often consists of components that are widely dispersed geographically, this might be of extra concern regarding remote access. Furthermore, IoT equipment is often equipped with physical interfaces (possibility to interact with the device itself, hands on). Such interfaces might provide challenges if and when such equipment is exposed to the public or nonauthorized users. Furthermore, such devices might be installed in places or areas without possibility to monitor the devices and by whom they are accessed.

6.2.2.3.4 Heterogeneity

IoT is typically cross-system, cross-product, and cross-domain. This often requires interoperability between heterogeneous components and systems. The degree to which heterogeneity is considered is often a matter of the protocols and communication needed for a product at different stages of the lifetime of the IoT system. The degree of heterogeneity for components and physical entities often drives cost and it is important that it is balanced compared with the business needs.

6.2.2.3.5 Highly distributed systems

IoT environments are often built by highly distributed systems which, while being functionally integrated, consist of sub-systems that can be physically separated and remotely located from one another. As an example, IoT systems can span whole buildings, whole cities and even the globe. Data can be stored at the edge of the network, centrally, or both. Distribution can also apply to processing – some processing taking place centrally or in the edge of the network, in IoT gateways or within more capable types of sensors and actuators.

It is important to consider the degree of distribution for the IoT system since most IoT systems are highly distributed by nature. Measurements often need to be taken to ensure the distributed components of such an IoT system can be handled remotely. This is needed for the component to be maintained, debugged, and upgraded and so on. Also, a high distribution might put constraint on trustworthiness characteristics of the system.

6.2.2.3.6 Modularity

Modularity is about how components can be removed cleanly from a system and replaced with another module of similar size and with similar physical and logical interfaces.

The modularity for an IoT system does not differ from the wanted modularity of any IT system. Nevertheless, it might be worth considering that IoT is a fast-growing technology. This makes it even more important to be able to upgrade separate parts of the IoT system continually; thus, placing even higher demands on the wanted degree of modularity. Modularity is also closely related to the composability characteristics of the system.

6.2.2.3.7 Network communication

IoT systems make use of a broad variety of network types. These can be limited range, low power networks collectively termed proximity networks that form the local connections for IoT devices. They can also be wide area networks that connect the proximity networks to the Internet. Gateways can be employed to connect networks of different types, typically between the proximity networks and the wide area.

The choice of communication protocols and network technology might be crucial to the business case and the life span of the IoT system. The choice of network technology also relates to the wanted scalability capability and data capabilities of the IoT system. Most communication protocols and network technologies often have both strengths and limitations. For crucial IoT applications the data characteristics of the chosen network technology are important.

Especially before investing in large scale implementation of sensors, it might be important to consider if the chosen network connectivity will be appropriate for the lifetime of the sensors.

It can also be beneficial to aggregate or in other ways reduce the amount of data transferred through the network.

6.2.2.3.8 Scalability

Since the number of connected entities in an IoT system often grows very quickly, it is of utmost importance that an IoT system is built to scale and to be able to handle a very large number of IoT devices as well as massive amounts of incoming data.

Furthermore, it is important for the user interface of an IoT system to consider how to display and sort information about millions of IoT devices, their data, and their metadata. The same goes for displaying a large number of rules and similar additional information.

Business managers might also want to take into consideration that deploying millions of IoT devices often will be at a large cost. Thus, the time it takes to deploy each IoT device and the possibility to mass deploy many IoT devices at the same time will have a large impact on the overall cost over time for the IoT investment as such.

When there is a risk of high load in incoming data it can be desirable to provide a means to prioritize incoming critical application sensor data.

New IoT actors sometimes can be at a developing stage where they primarily are working with demonstration and test data. It can be wise to carry out load tests with new IoT systems where the entire system is forced to handle a very large number of sensors simultaneously, be it real sensors or simulated.

6.2.2.3.9 Shareability

Many IoT components are underutilized since a single system often uses only a fraction of the capabilities of a component. If functionality or outputs of components can be shared among multiple systems, resources can be used more efficiently.

Determining shareability possibilities is often a good way to boost business opportunities for IoT components. At the same time business managers and system owners must ensure that a specific IoT component also meets the regulatory requirements. Especially for such other usage that differs from the primary intended functionality of that component.

6.2.2.3.10 Accuracy

Accuracy is a characteristic of various elements in an IoT system: Sensors make measurements on properties of the physical world. Accuracy of these sensors is the closeness of agreement between the measured values and the actual values of those properties. The usage of the data produced by the measurement is often highly dependent on the level of accuracy of the sensors as well as those computational algorithms used to manage the data. Actuators operating on the physical world translate digital commands into actions. The accuracy of such actions is also dependent on the accuracy of the provided data and the actuator itself.

It is often important to carefully decide what accuracy is needed for a specific business purpose. Often higher accuracy comes with a higher cost. IoT systems often consist of many components. Because of this, the impact on costs of choosing a component with higher accuracy than needed can result in unnecessary higher overall costs for the IoT system.

One thing to consider regarding accuracy is to what extent the surrounding environment might influence the accuracy. For example, a sensor measuring distance might be influenced by the air temperature or the temperature of the sensors, losing accuracy in high temperatures.

6.3 Business viewpoint and view

6.3.1 Business viewpoint

Table 3 describes the business viewpoint.

Table 3 – Business viewpoint

Viewpoint name		Business
Overview		Frames the concerns to be addressed by the business views
Known typical stakeholders		– Business manager – System owner – Architect
Concerns		– How to leverage the various capabilities of an IoT system to provide value for a business? – How to use IoT for innovative new business models? – How do characteristics of an IoT system influence business and system owner?
Viewpoint specification	Model kinds	Business model Service blueprint
	Legends	The business view provides text explaining the business implications of the essential characteristics described in the foundational view.
	View methods	NA
	Correspondence methods	NA
	References	NA

6.3.2 Business view

6.3.2.1 Business relevance of IoT

The Internet of Things is profoundly changing the world, by fundamentally changing the way that business is performed, and the way that value is delivered to customers. IoT can generate large volumes of data to give new insights, support new business models and lay the foundation for the delivery of products as services. More than perhaps any other technology, IoT impacts the overall business operation of companies and organizations. It is vital that the IoT implementations created today will lay a strong foundation for years to come.

This subclause 6.3.2 describes, in a nontechnical manner, some of the business aspects when creating or procuring IoT solutions. It also describes how to harmonize other IoT standards and guidelines with this document to avoid isolation effects and costs in the long run.

Making adequate technical decisions for IoT investments is crucial for business success.

This document can be leveraged by IoT stakeholders worldwide as organizations design and develop or acquire IoT solutions and services. The aim of this document is to provide a solid ground for long term and long-lasting investments for IoT. Suppliers are also likely to improve their ability to take part in bids on public sector and large corporate investments in IoT by following the guidance in this document.

From a business perspective, IoT systems offer several implications that can impact business models, operations, and revenue streams. Some of the key implications are as follows.

a) New revenue streams: IoT systems can enable businesses to create new revenue streams by offering value-added services such as remote monitoring, predictive maintenance, and real-time analytics. By leveraging data from IoT systems, businesses can offer insights and solutions that enhance the customer experience and generate additional revenue.

b) Operational efficiencies: IoT systems can improve operational efficiency by optimizing processes, reducing downtime, and streamlining maintenance. For example, sensors can provide real-time data on the performance of equipment, enabling businesses to schedule maintenance and repairs proactively, reducing costs and downtime.

c) Improved decision-making: IoT systems can provide businesses with real-time data and analytics, enabling better decision-making and more accurate forecasting. This can lead to better resource allocation, inventory management, and product development, among other benefits.

d) Enhanced customer experiences: IoT systems can enable businesses to deliver personalized and context-aware services, improving customer experiences and loyalty. For example, IoT-enabled devices can automatically adjust settings based on user preferences, enhancing the overall user experience.

e) Disruption of existing business models: IoT systems can disrupt existing business models, as they enable businesses to offer new products and services that were not previously possible. This can create new market opportunities and threaten the market positions of incumbents.

f) New partnerships and collaborations: IoT systems can enable businesses to forge new partnerships and collaborations, as they require expertise in multiple areas such as hardware, software, networking, and data analytics. This can lead to new business opportunities and the creation of new ecosystems.

Overall, IoT systems offer significant implications for businesses. However, as the adoption and implementation of IoT systems can be complex and require significant investment in hardware, software, and personnel, businesses will typically evaluate the costs and benefits of IoT systems to determine whether they align with their strategic goals and objectives.

6.3.2.2 Business implications of IoT systems

Business managers and system owners are stakeholders in the IoT RA. They have various interests in the system of interest (represented as concerns). These concerns are addressed in the views of the reference architecture description. The business concerns can be relevant to anyone interested in business management or ownership aspects of IoT. They give aid and guidance to support:

- producing or acquiring IoT solutions and IoT services;
- integrating IoT with existing IT solutions;
- planning for the future of existing IoT products and services;
- reshaping business models to take advantage of IoT capabilities.

6.3.2.3 IoT RA and other IoT initiatives

There are many IoT standards, guidelines, and initiatives in existence today. For their given application field and purpose, most of these standards and guidelines work well on their own and are used by various IoT stakeholders. As a result, there is a diverse and heterogenous IoT world today. Heterogeneity is an important aspect of IoT. Nevertheless, interoperability and convergence between IoT standards and guidelines are necessary to support the combination and interaction of these heterogeneous resources.

Uncertainty about resource compatibility makes it difficult for the stakeholder decisions about both a foundation for long-term investments and durable protection of current cornerstones.

This document serves as a foundation for creating interoperability and alignment between IoT initiatives. The aim of this document is to bring different views together.

6.4 Usage viewpoint and view

6.4.1 Usage viewpoint

Table 4 describes the usage viewpoint.

Table 4 – Usage viewpoint

Viewpoint name		Usage
Overview		Framing of the concerns to be addressed by the usage view
Known typical stakeholders		– System architect – Project manager – Programme manager
Concerns		– How do users (both human and digital) interact with the IoT system? – How does the IoT system interact with the physical entity of interest?
Viewpoint specification	Model kinds	Usage model Activity model Operational mode
	Legends	The view provides text describing both the interactions between the users and the IoT system and the interaction between the IoT system and the physical entity of interest.
	View methods	NA
	Correspondence methods	NA
	References	NA

6.4.2 Usage view

6.4.2.1 External interaction model

IoT systems interact with human users, other digital systems (including digital users), and the physical world. Users interact with the IoT system to gain new insights about or control over an entity of the physical world. Understanding what those interactions are is critical to designing IoT systems.

The external interaction model (Figure 4) addresses the interaction of the IoT system with users (both digital and human) and the IoT system's interaction with the physical world (through sensing and actuating):

- physical interaction;
- interfaces.

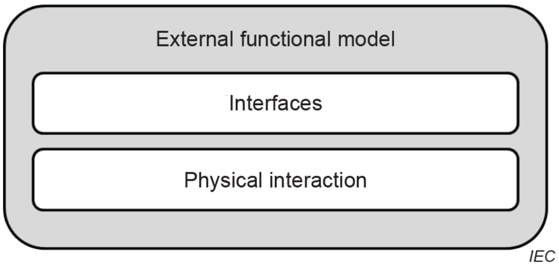


Figure 4 – External facing functions

The user interaction provides user access to the primary functions of the IoT system.

In this context, the primary function refers to the capabilities made available by the IoT system, including:

a) services of various natures (application, operations, and management, brokering with other external resources including data, information, or services);
b) data and information from sensors or controllers, including events and notifications;
c) contextual business data;
d) derived knowledge and information that is generated by applications and services in the IoT system such as intelligence created by analytic processes.

6.4.2.2 Physical interaction

The physical interaction consists of common functional groups for interacting with entities in the physical world through sensing and actuating. The implementation complexity depends on the infrastructure of the specific IoT system being implemented.

6.4.2.3 Sense functions

Sensing is the function that reads sensor data from sensors. Sensing implementation spans hardware, firmware, device drivers, and software elements. For example, an attention element to tell the sensor what is needed.

6.4.2.4 Actuate functions

Actuation is the function that writes data and control signals to an actuator to affect the actuation. Its implementation can span hardware, firmware, device drivers, and software elements. It is local in the sense that it closes loops (logically) close to sensors and actuators. These can still be physically co-located with other central resources if resilience and performance requirements can be met.

6.4.2.5 Interfaces (user interactions)

For machine users, there are interfaces through which the capabilities of the IoT system can be invoked over the network. These are application programming interfaces (APIs), and portal functions provide controlled ways of accessing IoT system functions, either by digital or human users. Digital users normally interact through APIs, and human users through access portals.

For human users, there are applications which offer user interfaces that enable interaction with capabilities of the IoT system. For human users, the user domain also contains the end user devices which support the applications.

The interfaces include:

- resource interfaces – for services – and their life cycle management.
- resource discovery, publishing, search, and querying.

6.5 Functional viewpoint and view

6.5.1 Functional viewpoint

Table 5 describes the functional viewpoint.

Table 5 – Functional viewpoint

Viewpoint name		Functional
Overview		The functional view provides an understanding of the types of logical functions found in IoT systems.
Known typical stakeholders		– System architect – Project manager
Concerns		– What are the types of abstract functions to be implemented in an IoT system?
Viewpoint specification	Model kinds	Function model Feature model Capability model
	Legends	The functional view provides text describing the three abstract categories of functions found in IoT systems: management, communication, and data.
	View methods	NA
	Correspondence methods	NA
	References	NA

6.5.2 Functional view

6.5.2.1 Internal functional model

Once the system interactions are captured, an understanding of what abstract functions are needed to design a system that can perform those interactions is needed. These abstract functions can then be instantiated using the appropriate construction patterns.

The internal functional model is a technology-agnostic set of functions that are found in an IoT system. Each box in the diagram represents a type of function that is found in an IoT system. Each type of function can be realized by one or more different implementations of system components, which are deployed to form a working system. It does not go to a detailed level, leaving flexibility for implementation covered in the implementation view. The functional view only addresses the types of abstract functions which are essential for every IoT system. It is up to the developer to decide how to build the functional components to implement these abstract function types. The functional model is composed of three basic function types: management, communication, and data (see Figure 5).

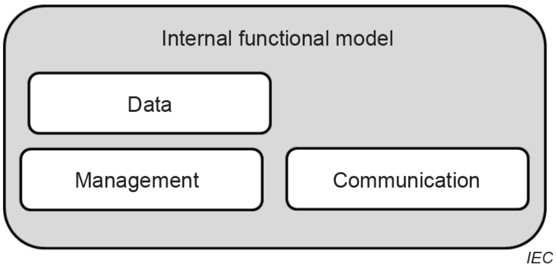


Figure 5 – Internal model of abstract function classes

6.5.2.2 Data function

The data function types can be broken down into three basic function types – control, management, and analysis. Control and analysis are both types of data processing, while data management focuses on how data relate to other data, systems, and the physical world.

6.5.2.3 Control functions

Control functions exist to control local state – in particular to issue commands to actuators based on input from sensors and other sources. It is common for control services to have real-time behaviour because of the need to control dynamic elements in the PED; both to ensure appropriate operation of the system and to ensure safety of operation.

6.5.2.4 Analysis functions

Analysis functions represent the functions implementing application and service logic that realizes specific business functionalities. Analysis covers analytic services, cognitive services, streaming data services, visualization services, business rules services, and application logic.

Analytics services: Gathered data (sensor data streams, other context data and internal system state) are processed to create insights. Analytic services can act on real-time events and historical data.

6.5.2.5 Data management functions

Data management functions include functions for operation on data which are collected by the IoT system. Those operations focus on how data relate to other data, systems, and physical world. Such operational functions include data transformation, filtering, cleansing, ETL (extract, transform, load), rules and notifications. The data management functions also contain data stores of various kinds including a device data store, analytics data store, and historical data store.

6.5.2.6 Management function

The management function types contain functions responsible for the overall management of IoT components, systems, and environments.

Management function includes functions for the operational management of the IoT system. Management function includes provisioning, monitoring, reporting, interaction between components, policy management, service automation, and service level management. Management function also includes service catalogues, device registries, device management. There is also a need for management functions for business aspects of the IoT system, including account management, subscription management, billing accounts.

Application support provides the execution infrastructure that the components deployed in the IoT environment can use to achieve, for example, scalability and configurability. It also provides the tools that are required by a service or application to do accounting and billing.

Business services help create business process flows, and orchestration of resources to create and manage services. The management function should consider cybersecurity in accordance with the guidance of the IEC 62443 series [4] and ISO 27000 [5].

IoT services can contain discovery functions which enable access to appropriate capabilities within the IoT system for external and internal users. Such resources are typically applications, services, and data, but can also include administration capabilities and business capabilities. Such management-related functions include:

- metadata management and usage;
- accessing and managing directories and repositories.

6.5.2.7 Communication function

IoT systems are composed of components that interact over many-to-many networks. These many-to-many networks provide the data transfer capability to move data from one component to (potentially) any other component. It is also the connection between components that enables those components to interact. The communication function can be broken down into three categories: the data transfer function itself, the network interface functions, and the application interface functions.

The characteristics of this network have a drastic impact on the performance of the IoT system. Communication network types typically used for IoT systems are characterized as follows.

- Proximity networking: Proximity networking enables transmission of data from assets or devices on the edge to entities such as gateways. Gateways and similar entities can then process this data for further transmission, or for enacting controls, such as in edge or fog computing. This networking also enables the control of assets, by actuators or controllers. Proximity networking is often limited to the SCD, but in some occurrences, proximity network and access network are the same network.
- Access networking: This networking function enables the transfer of edge data to application logic or operations logic. It also enables the communication of control signals to interacting (sensing and actuating) entities. It is supportive of management and higher-level communication functions.
- Service networking: Service networks are enabling service-based deployment of IoT applications, for example using microservices and other shared services.
- User networking: This networking function gives both human and digital user entities access to and control of IoT systems. It also enables higher levels of integration between different IoT systems and with non-IoT systems by supporting user-facing entities.

6.5.2.8 Security function

Security functions in the IoT system ensure the confidentiality, availability, integrity, and authenticity of information. The IoT RA integrates security policies for IoT components as a key part of system design.

6.5.2.9 Resilience function

Resilience functions enable the IoT system to recover to the operational condition quickly following an incident. Resilience functions are closely related to autonomic computing capabilities of self-healing, self-configuring, self-organizing, and self-protecting.

6.5.2.10 Reliability function

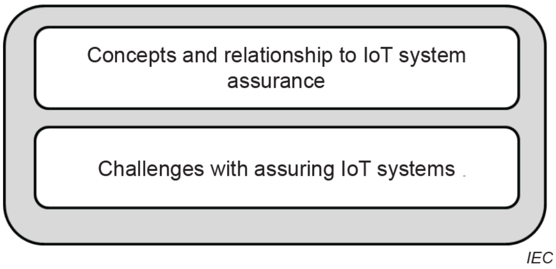
Reliability functions ensure that the IoT system, or a component in the IoT system, can perform its required functions under stated conditions for a specified time.

6.6 Trustworthiness viewpoint and view

6.6.1 Trustworthiness viewpoint

Table 6 describes the trustworthiness viewpoint, taking into account ISO/IEC TS 30149 [6].

Table 6 – Trustworthiness viewpoint

Viewpoint name		Trustworthiness
Overview		Framing of the concerns related to trust, confidence, and assurance of an IoT system.
Known typical stakeholders		– System architect – Security engineer – Security manager – Privacy manager – Project manager – Business manager
Concerns		– How to design an IoT system with a level of confidence that meets trustworthiness goals. – How to assure the implemented system meets design goals.
Viewpoint specification	Model kinds	NA
	Legends	The trustworthiness view provides text describing the concept of trustworthiness and its relationship to IoT system assurance and specific challenges associated with assuring IoT systems as shown in Figure 6.  Concepts and relationship to IoT system assurance Challenges with assuring IoT systems IEC **Figure 6 – Legend used in the trustworthiness view**
	View methods	NA
	Correspondence methods	NA
	References	NA

6.6.2 Trustworthiness view

6.6.2.1 Concepts and relationship to IoT system assurance

Trustworthiness is an important concept in IoT systems because system failure has real-world consequences. This challenge is made more difficult by the separation of resource between information technology (IT) and operation technology (OT). When an IoT system is being designed and built, stakeholders need to know the system is trustworthy.

Trustworthiness and assurance are two related but distinct concepts. Trustworthiness refers to the quality of being reliable, honest, and dependable, while assurance refers to the measurable level of confidence that something will perform as expected or meet specific requirements.

In the context of system engineering, trustworthiness refers to the extent to which a system can be trusted to operate as expected and meet both functional and non-functional goals. Trustworthiness involves a range of factors, including reliability, security, resilience, safety, environmental impact, ethics, and others.

Assurance, on the other hand, refers to the measurable level of confidence that a system will perform as intended and meet specific requirements based on the trustworthiness goals. Assurance is achieved through various means, such as testing, validation, and certification, and is done both internally and through independent third-party organizations that evaluate and certify a system.

6.6.2.2 Challenges with assuring IoT systems

As IoT systems are complex systems, existing system engineering practices for system assurance and risk mitigation can be leveraged. An IoT system will exhibit the essential characteristics that are described in 6.2.2, the foundational view of this document. The implications of these characteristics on system trustworthiness for that system should be considered by the system architect. The many-to-many networking characteristic indicates that any component can connect to any other component, and any functionality can be put behind a network interface and used as an IoT component (including another IoT system). Therefore, the potential structure of IoT systems is limitless. This variation in architecture is addressed in this document within the construction view and the use of patterns.

The components in an IoT system are connected digitally instead of using nuts and bolts, and such systems can usually be built faster than a physical system. The system structure can even be changed during run time. This should be considered when designing IoT systems.

The IT characteristics of IoT systems, such as the nondeterministic nature and shared aspect of these systems, create challenges in assuring IoT systems. Unlike a physical device that is disassembled to access individual components, an IoT system's individual components are easy to access, and this ease of access presents unique security challenges.

The physical nature of IoT systems and the interaction of these systems with the physical world provide new challenges for assurance as compared to purely digital systems. Measurements and observations have uncertainties that do not exist in information systems. The actions performed by an IoT system, resulting in a change to the physical world, have tolerances that should be considered by the architect. IoT systems with actuators introduce the possibility of physically damaging equipment and facilities, and harming people. The nature of errors in digital systems creates new challenges for system architects, as the consequences of one single flipped bit can cause catastrophic failure in the physical world.

These challenges are exacerbated by the structure of enterprises and the typical organizational division between operational technology experts and information technology experts. This division creates specific challenges that are related to safety, security, privacy, reliability, and resilience.

6.6.2.3 Characteristics

6.6.2.3.1 Availability

Availability covers both the availability to functions and services within an IoT system or IoT environment as well as the continued access to vital data. Availability of a device is related both to its inherent properties of operating correctly over time and to the network connectivity of the device. Availability of data is the ability of the system to get the requested data from a system or component. Availability of services is related to the ability of the system to provide the requested service to users.

When an organization starts to use IoT as a base for decisions or automated actions, it becomes dependent on the technology. The organization will rely on the availability of data, devices and services delivered by the IoT system. It is important to understand what IoT data or services must be available and the consequences if they are not available.

6.6.2.3.2 Confidentiality

Confidentiality is the property that information is not made available or disclosed to unauthorized individuals, entities, or processes. In an IoT system, protection policies and mechanisms are responsible for prohibiting people or systems from accessing data or control messages when they are not authorized to do so.

IoT data often have one meaning and information classification for a given purpose. They can have another meaning in another context or when combined with other data, thus resulting in a higher wanted security classification. Also, a specific dataset might be harmless when accessed alone but harmful when combined with other data. Therefore, it is important to analyse what possible threats can be caused by various information breaches in the IoT environment; even for IoT data that are otherwise considered as simple.

6.6.2.3.3 Integrity

Integrity is accuracy and completeness applied to information within a system. Integrity is vital for IoT systems to ensure that the data used for decision-making processes in the system and executable software have not been altered by faulty or unauthorized devices, by malicious actors, or by environmental causes.

Not only can corrupt data cause malfunctions. Unauthorized manipulation of data can also compromise the trustworthiness and business brand of the organization; especially if such information is displayed in public or shared with other organizations. When IoT is used for new business models and new working methods, the business owner should consider the integrity risks. How to avoid them or detect them and what consequences any breaches might have on business and work progress.

6.6.2.3.4 Reliability

Reliability is a property of consistent, intended behaviour and results. An appropriate level of reliability is essential in diverse IoT system deployments and applications. Reliability can be highly critical in some applications, as for specific health-related applications, industrial manufacturing operations, and time-critical applications. The complexity of many IoT systems, often with many separate IoT components from various vendors, places even higher demands on addressing the overall IoT system reliability.

Business owners should bear in mind that all components in the IoT system make up the overall reliability. The weakest component might define the level of reliability of the whole system.

6.6.2.3.5 Resilience

Resilience is the ability of an IoT system or its components to adapt and continue to perform their required functions flexibly in the presence of faults and failures and other ad hoc changes without loss of operation and performance level. It is important for IoT systems to be designed for resilience, incorporating self-monitoring and self-healing techniques to improve the system resilience.

It is important to consider how the overall resilience of a product or larger IT, OT and IoT systems might be influenced by the level of resilience for specific IoT components. One important aspect to consider is if it is acceptable to lose collected data during a recovery phase. Otherwise, actions should be taken to preserve such data during recovery.

Another aspect to consider is what down time is acceptable and what actions to take to ensure an appropriate recovery time frame.

6.6.2.3.6 Safety

Safety is the state in which the risk of harm of the health of people, damage to property or the environment is limited to an acceptable level. Risk level is the probability of the occurrence of harm combined with the severity of that harm. Safety standards often must be considered in IoT systems including medical or health care, transport such as aviation and automotive applications, consumer products, buildings, environment monitoring, fire safety, national border safety, and radiation damage monitoring.

Even in contexts where compliance with safety standards is optional or voluntary rather than mandatory, proper consideration of safety factors is important. This can have significant impact on aspects such as continuity of operations, reduction of loss, prevention of injury or death, insurance premiums, torts and liability, and other issues.

6.6.2.3.7 Compliance

Compliance is conforming to rules, such as those defined by a law, a regulation, a standard or a policy. IoT systems, services, components, and applications can be deployed in circumstances which require adherence to a variety of laws, policies, or regulations.

Regulations of relevance to IoT systems can take many forms. These include regulations to assure interoperability, to mandate or constrain functionality or capability, to assess the ability of the IoT device or system to function in a certain usage context without causing damage, and to impose at least minimal balance between contribution to the collective good and self-interest on the part of system owners or operators. It is important to pay extra attention to compliance issues when using an IoT component for several purposes. The same is true when an IoT component is used for purposes other than the component has initially been produced and designed for. As provider of IoT technology it is often important to clearly state the compliance for all parts provided. Using IoT components lacking compliance assurance for an IoT system might cause legal breaches and can result in heavy penalties. It can also cause the IoT system to fail and put business, environment, people, and more at risk. Furthermore, such breaches might cause legal business controversies and insurance uncertainties.

NOTE Some more examples of IoT characteristics with relevance for trustworthiness and assurance can be found in Clause C.2.

6.7 Construction viewpoint and views

6.7.1 Construction viewpoint

Table 7 describes the construction viewpoint.

Table 7 – Construction viewpoint

Viewpoint name		Construction
Overview		What are the specific IoT system structures for addressing different requirements?
Known typical stakeholders		– Project manager – Programme manager – Standards expert – System owner – Business manager
Concerns		– What types of design are useful when creating an IoT system to meet a given set of requirements? – What types of design are useful when creating an IoT component to meet a given set of requirements?
Viewpoint specification	Model kinds	NA
	Legends	The construction view uses the construction pattern legend described in Table 8.
	View methods	Construction views are specified using the construction pattern.
	Correspondence methods	NA
	References	NA

Table 8 – Construction pattern legend

Information	Name	Name of pattern The pattern's name conveys the essence of the pattern succinctly. A good name is vital because it will become part of the working vocabulary.
	Related patterns	Similar patterns, depending patterns. There can be similar patterns. The pattern can extend other patterns.
Problem		Description of problem which the pattern attempts to solve. A short statement that answers the question: What particular issue or problem does it address?
Known Context	Specific context	The particular context in which the pattern solves a problem. Where does the pattern apply? For example, the use of an enterprise-wide data model frequently makes sense in a problem context where distributed data management is a concern while the architecture for an air-to-air missile will not be an appropriate context for this pattern.
	Related context	Other related context
Solution	Architecture models	Architecture models for the pattern Text and diagrams necessary to understand the essential concepts and relationships for the pattern
	Examples	Scenarios/use cases where the pattern has been applied Useful patterns are motivated by known, previous usages Examples (and visual analogies) help explain the pattern
	Rationale for the pattern	The rationale can be theoretical (e.g. the mathematical theory of rate monotonic scheduling) or practical (e.g. prior case studies in which the pattern was successfully employed
	Guidance	Provide useful information that assists an architect in using the pattern. Guidance can include: – Description of characteristics – References to other documents (e.g. standards, regulations, white papers, ontologies) – Discussion on pain points, critical decisions, underlying requirements, trade-offs

6.7.2 Construction view

The construction view is used in order to construct an IoT architecture which is further constraining the IoT reference architecture through to the integration of specific construction patterns.

EXAMPLE 1 The RAMI 4.0 architecture [7] considered as a pattern is integrated to an IoT reference architecture to support the smart manufacturing domain. RAMI 4.0 is described in Annex A.

EXAMPLE 2 THE SGAM architecture pattern [8] considered as a pattern is integrated to an IoT reference architecture to support the smart grid domain.

EXAMPLE 3 An implementation architecture for industrial IoT can use the patterns library in [9].

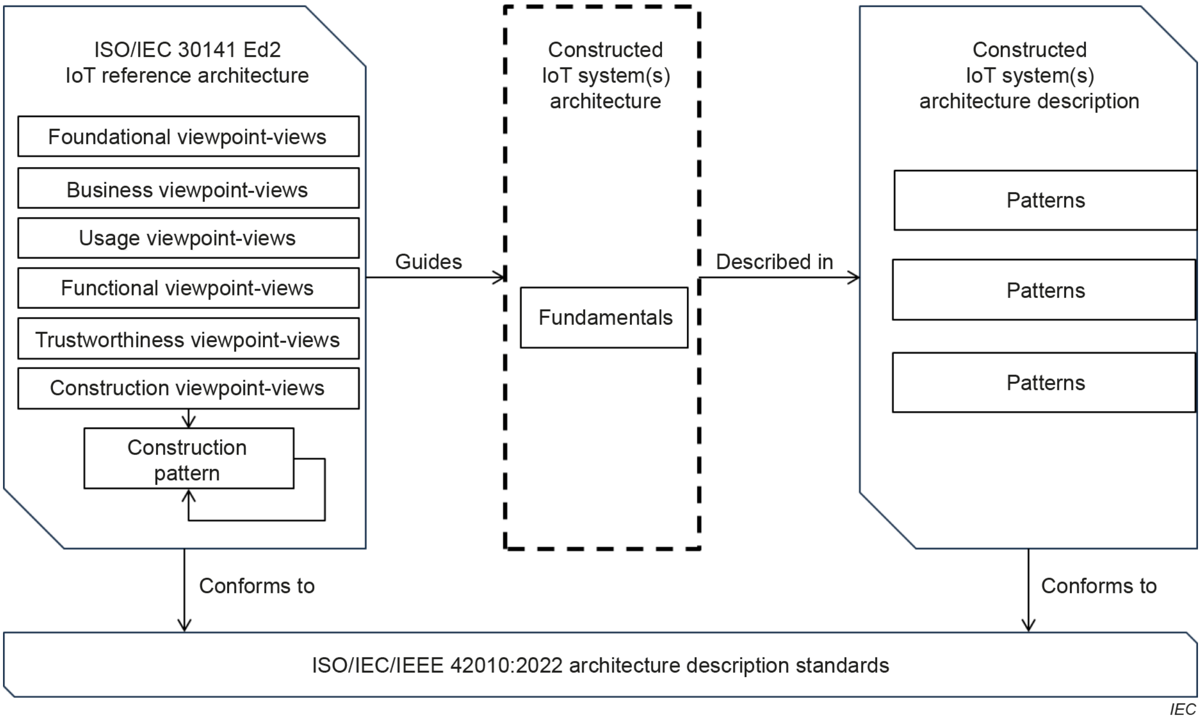

ISO/IEC 30141 Ed2
IoT reference architecture
Foundational viewpoint-views
Business viewpoint-views
Usage viewpoint-views
Functional viewpoint-views
Trustworthiness viewpoint-views
Construction viewpoint-views
Construction pattern
Guides
Constructed IoT system(s) architecture
Fundamentals
Described in
Constructed IoT system(s) architecture description
Patterns
Patterns
Patterns
Conforms to
Conforms to
ISO/IEC/IEEE 42010:2022 architecture description standards
IEC

Figure 7 – IoT architecture construction view

Figure 7 shows the construction view:

- The left part provides a representation of viewpoints and views of this document, with the construction viewpoints and view which is based on the use of construction patterns.
- The right part provides a representation of the resulting viewpoints and views of the constructed IoT architecture: the business, usage, functional and trustworthiness viewpoints and views are extended and specialized with patterns.

The construction of an IoT architecture follows the following view method:

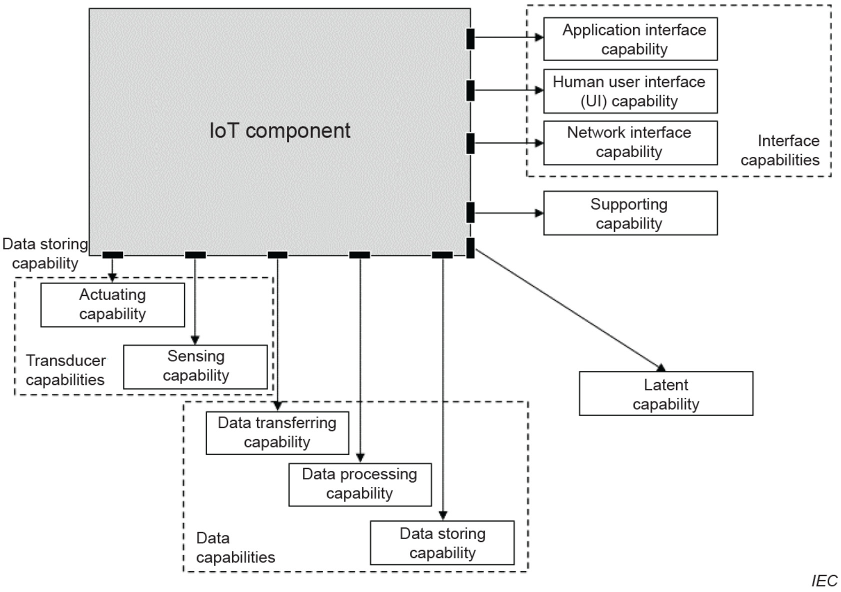
- The starting point are the views of this document.
- The views are extended or specialized through additional architecture models which further constrain the views of this document.
- The specification of the additional architecture models is based on patterns.
- Several reference architectures can also be combined by considering a reference architecture as a pattern [9].

  EXAMPLE 4 The IoT reference architecture is extended to support digital twin capabilities by considering a digital twin reference architecture as a pattern.

6.7.3 IoT component pattern

The construction view method can involve the use of the IoT component pattern, described in Table 9.

Table 9 – IoT component pattern

Information	Name	IoT component
	Related patterns	-
Problem		Understanding the set of capabilities of a given IoT component is essential to being able to use the component in an IoT system with confidence. A simple definition of capability is "the quality of being able to perform a given function".
Known Context	Specific context	Many IoT components do not make all their content visible to others. Organizations acquiring, using, and administering them have little or no access to information about their internal workings, including the capabilities that they offer. For other IoT components, with detailed information about the internal workings, there are often no standardized mechanisms to expose, access, and configure capabilities.
	Related context	-
Solution	Architecture models	The diagram that is shown in Figure 8 provides a model of the capabilities an IoT component can provide. In other words, each IoT component can be characterized by the set of capabilities it provides. A given IoT component can have more than one of any given capability type (sensors, network interfaces, actuators). This model can also be used to describe the set of components and relationships of a given IoT system or the capabilities of an IoT system.  IoT component Application interface capability Human user interface (UI) capability Network interface capability Interface capabilities Supporting capability Data storing capability Actuating capability Transducer capabilities Sensing capability Latent capability Data transferring capability Data processing capability Data capabilities Data storing capability IEC **Figure 8 – Capabilities of an IoT component**
	Examples	-
	Rationale for the pattern	Provides a refinement of IoT component categories of capabilities (transducer, data, interface, supporting, latent)

	Guidance	The IoT capabilities can be grouped into several categories: – Transducer capabilities interact with the physical world. These capabilities, prevalent in OT, serve as the boundary (edge) between the digital and physical environments. Transducer capabilities provide the ability for computing systems to interact directly with physical entities of interest. – Data capabilities are directly involved in providing functionality to the system. These capabilities—data storing, transferring, and processing—are commonly associated with conventional IT systems, and are critical to IoT systems. – Interface capabilities provide the component with the ability to interact with other IoT components (including people using a connected device to interact with the other system components). – Supporting capabilities are indirectly involved in providing functionality to the system, such as monitoring, management, security, or orchestration. – Latent capabilities are transducer, data, interface, or supporting capabilities that are not currently enabled and accessible outside the IoT component. These capabilities can potentially be enabled either by a trusted actor or a bad actor with malicious intent.

Table 10 provides additional information about each capability category.

Table 10 – Additional information on IoT component capabilities

Transducer capabilities	Actuating: An actuating capability, provided by an actuator, offers the ability to change the physical world. Such change is based on information that is given as input to the component. Errors can be introduced in the digital logic, the digital-to-analogue converter, the analogue electrical circuit, and the actuator transducer. There is a time delay between the input data arriving at the component and the change being made to the environment. Examples of actuating capabilities include heating coils (heating capability), electric shock delivery (cardiac pacing), electronic door locks (lock/unlock capability). Other examples are unmanned aerial vehicle operation (remote control), servo motors (motion/movement capability), and robotic arms (complex motion/movement capability). An important type of actuator is a black box control system that accepts a wanted outcome as an input and internally uses sensors, actuators, and processors to make the physical changes. Such a black box control system is considered an actuating capability in this model since the sensors and processors are not directly usable from outside the component. Sensing: A sensing capability (provided by a sensor) offers the ability to provide observations of an aspect of the physical world as measurement data. Information from sensor observations can be provided to other IoT components through the network interface of the component for processing and storage. Sensing is "read only". Any change to the physical state is a side effect. Measurement errors can be introduced by the physical environment between the physical system and the sensor transducer and in the sensor transducer itself. Measurement errors can also be introduced in the analogue electrical circuit, in the analogue-to-digital (A/D) converter, and in the digital logic of the sensor. There is also a time delay between the sensing and the data becoming available at the component output. Examples include temperature sensing (temperature measurement capability), computerized tomography (CT) scans (radiographic imaging), spatial sensing (accelerometers, gyroscopes), optical sensing, and audio sensing.

Data capabilities	Data storing: A data storing capability provides ability to store and retrieve data and information over time. The intent is to store data for use at some later time. Data persists for a finite period. Data can be published by the component or provided in response to an external request. There is a time delay between the input and output, that is between a data request and the data response. Examples of data storing capabilities include databases and data brokers. Data storing capabilities can also be any other type of component that stores input data for later use. Data transferring: A data transferring capability provides the ability to transmit data from one physical or logical location to another. The data transferring capability provides ability to provide information about the network without having to understand the specific network topology. The interactions of an IoT system with the physical world require the data transferring network to meet latency, reliability, and security requirements. Therefore, it is useful to be able to describe the network characteristics in this manner, so the capability is explicitly called out by the IoT general model. Examples of specific data transferring capabilities include data networks that are based on Ethernet, Wi-Fi, LTE, Foundation Fieldbus, Modbus and BACNet. Data processing: A data processing capability provides the ability to transform data based on an algorithm. The intent of processing is to transform input data and provide output data. There is a time delay between the input and output that should be accounted for. The transformation can be simple, with a single input variable, and a single output, or it can be complex with multiple inputs and outputs. Control algorithms are an important type of processing. Control algorithms take the output of sensors and actuators or pre-processors and provide an output that can be fed into an actuator or post-processor. These control algorithms are often used within negative feedback loops, but not always. A proportional-integral-derivative (PID) control algorithm is an example of such a control algorithm. Some examples of processing include data aggregation, binary (Yes/No) analysis, big data analytics, machine learning, and predictive analysis.

Interface capabilities	Application interface: An application interface capability provides the ability for other IoT components (components, systems, etc.) to communicate with a given IoT component through an IoT component application. A widely used type of application interface is an application programming interface (API). Human user interface: A human user interface (UI) capability provides the ability for the component to communicate directly with people. Not all IoT components have a human UI capability (that is, a dedicated processing component). Any effect on the physical environment is a side effect of the interface (the purpose being information exchange). Such an effect is not considered to be sensing or actuating. Examples of human UI capabilities include keyboards, mouses, microphones, cameras, scanners, monitors, touchscreens, touchpads, speakers, and haptic devices. Network interface: A network interface capability provides ability to interface with a digital communication network for communicating data from one component to another. Every IoT component has at least one network interface capability and can have more than one. While the network interface capability allows for a component to be connected to a communication network, it does not provide the communication (data transferring) capability. Some examples of network interface capabilities include Ethernet adapters, LTE radios, ZigBee radios, Fieldbus and Wi-Fi dongles. Supporting capabilities: Supporting capabilities provide additional functionality that supports the IoT system. Examples of supporting capabilities include time synchronization, data encryption, authentication, orchestration, and remote component management. Some IoT components can only provide a supporting capability such as orchestration and not offer any transducer or data capabilities. Latent capabilities: The latent capabilities are capabilities that the IoT component can potentially provide but are not currently enabled for access externally from the IoT component. For example, a component can have an empty USB port with nothing plugged into it. In that state, the USB port is considered a latent capability. It has the potential to be used at any time. If someone attaches something to it, that can enable any of the other capabilities. If someone plugs a Wi-Fi adapter into the USB port, the IoT component would then have an additional network interface capability. USB ports and other communication interfaces, such as serial, High-Definition Multimedia Interface (HDMI), Digital Visual Interface (DVI), DisplayPort, and External Serial Advanced Technology Attachment (eSATA), often change their state. They can then switch from being a latent capability to an active capability and back.
NOTE Any tradenames and trademarks in this table are examples of suitable products available commercially. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of these products.	

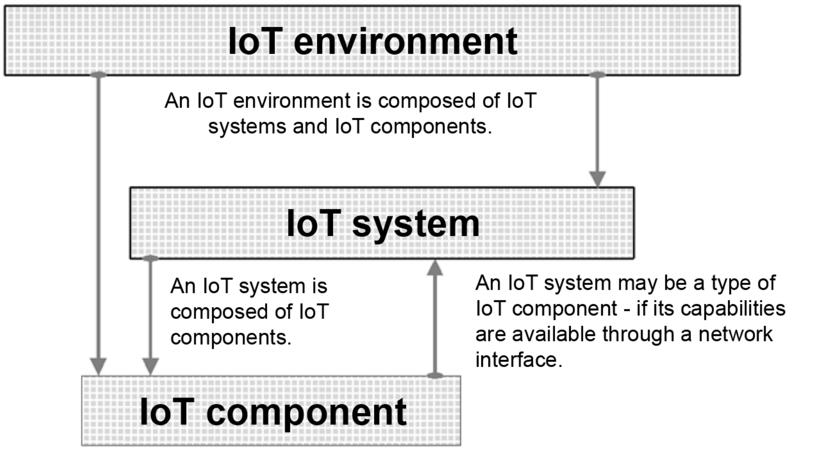
To provide value or benefit to an IoT system, an IoT component can perform some type of transformation. Key capabilities and their respective transformations are listed in Table 11.

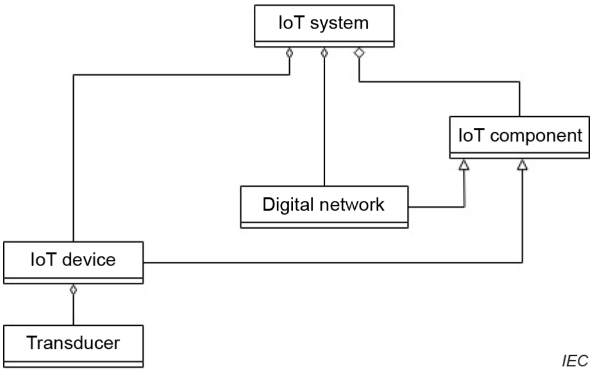
Table 11 – Key capability transformations

Capability type	Input type	Transform input	Transform output	Output type
Sensing	Physical energy	Property of physical system state	Representation of property of physical state	Digital data
Actuating	Digital data	Representation of wanted change in aspect of physical state	Changed property of physical system state	Physical energy
Data processing	Digital data	Set of information	New set of information	Digital data
Data storing	Digital data	Set of information	Set or subset of information available over time	Digital data
Data transferring	Digital data	Set of information	Same set of information available over distance	Digital data

The IoT system pattern is described in Table 12.

Table 12 – IoT system pattern

Information	Name	IoT system pattern
	Related patterns	IoT component pattern
Problem		
Known context	Specific context	The IoT system pattern applies to all IoT systems
	Related context	-
Solution	Architecture models	Relationship between an IoT system and its components and environment.  IoT environment An IoT environment is composed of IoT systems and IoT components. IoT system An IoT system is composed of IoT components. An IoT system may be a type of IoT component - if its capabilities are available through a network interface. IoT component IEC IoT systems are composed of: – one or more networking components – two or more additional components – one or more components must interact with the physical world through sensing or actuating. IoT systems can provide the following service capabilities. Primary: • physical observations and derived information based on those observations • control of physical entities Secondary: • data processing • data transferring • data storage Interfaces • network • human UI • application Supplemental: • security • orchestration • management

		IoT system class diagram  IoT system IoT component Digital network IoT device Transducer IEC
	Examples	A IoT-powered cycling app combines real-time data from sensors on the user's cell phone, including GPS, accelerometers, and heart rate monitors, with Internet connectivity to offer cyclists comprehensive ride tracking and performance analysis. The user's cell phone serves as both a display UI and a data collection device, while cloud servers store and process the data. The app provides cyclists with real-time feedback on metrics like speed and distance, tracks and maps their rides, and offers historical data for performance comparison. It also integrates with external data sources and enables social sharing, enhancing the overall cycling experience by improving performance and safety. A smart grid is a modernized and interconnected electrical grid system that harnesses IoT technologies to optimize electricity generation, distribution, and consumption. It deploys sensors and measurement devices across the grid infrastructure, connecting them through robust communication networks. These sensors continuously collect data on parameters like voltage, current, and power quality, enabling real-time monitoring and control from central control centres. Advanced data analytics processes this information, empowering grid operators to make informed decisions, predict equipment failures, and enhance energy distribution efficiency. IoT-enabled demand response programmes, distributed energy resources integration, automation for fault detection and grid resilience, and cybersecurity measures all contribute to the smart grid's efficiency, reliability, and sustainability.
	Rationale for the pattern	This is the primitive pattern that all other IoT system patterns are based on.
	Guidance	Guidance for IoT systems applies. Guidance for IoT devices applies. Guidance for IoT networks applies.

Annex A
(normative)

Additional IoT construction patterns

A.1 General

This Annex A describes the following patterns:

– the Reference Architecture Model Industrie 4.0 (RAMI 4.0) pattern;

– the IoT user pattern;

– the dynamic IoT system pattern;

– the IoT enterprise system pattern;

– the IoT enterprise networking pattern, which uses the IoT enterprise system pattern;

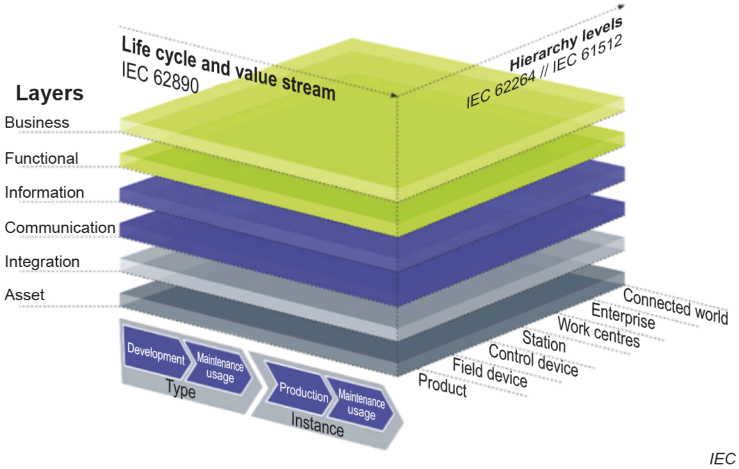
– the IoT enterprise usage pattern.

A.2 Reference Architecture Model Industrie 4.0 (RAMI 4.0) pattern

The Reference Architecture Model Industrie 4.0 (RAMI 4.0) pattern is described in Table A.1.

Table A.1 – RAMI 4.0 pattern

Information	Name	RAMI 4.0 pattern
	Related patterns	-
Problem		Reference Architecture Model Industrie 4.0 in accordance with IEC PAS 63088 [7]. This model describes all needed information about any object with all aspects for mirroring that object from the physical world into the information world in a cubical way. The pattern allows mirroring an object from the physical world into the information world. Mirroring is done by a minimum set of characterizing aspects with sets of properties (data elements) in accordance with IEC 61360 [10].
Known context	Specific context	Context: The pattern has been developed in the context of Industrie 4.0 / smart manufacturing but can also be an example for smart cities and other smart worlds. Typical scenarios: The basic scenario is the mirroring of an object from the physical world into the information world. Examples for the objects are ideas, patents, software, plans, services, processes and so forth in the IoTS (Internet of Things and Services). Typical problems addressed by the pattern: The pattern is the core component in the future smart worlds.
	Related context	-

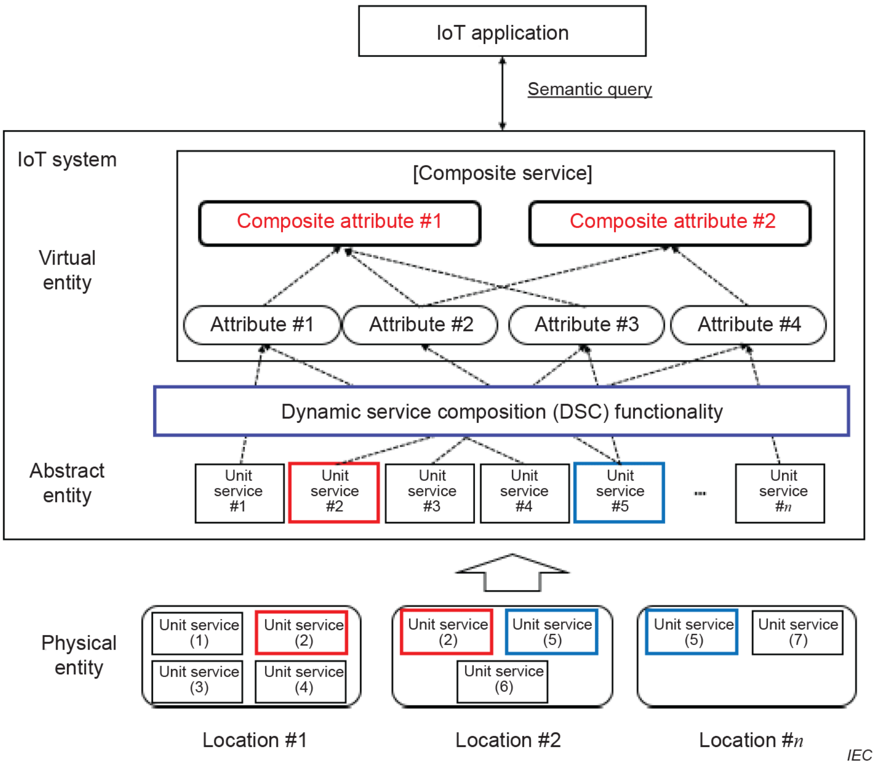
Solution	Architecture models	Following is the Reference Architecture Model for Industrie 4.0 (RAMI4.0). The full RAMI 4.0 specification is included in IEC PAS 63088 [7]. It shows all relevant information that is needed to represent an object in the information world.  Layers Business Functional Information Communication Integration Asset Life cycle and value stream IEC 62890 Hierarchy levels IEC 62264 // IEC 61512 Connected world Enterprise Work centres Station Control device Field device Product Development Maintenance usage Type Production Maintenance usage Instance IEC **Figure A.1 – RAMI 4.0** Figure A.1 shows the RAMI 4.0 layers. In the smart world all relevant information of the information world is available in a computer readable format. The information includes a standardized name in a standardized binary format referring to the asset and its standardized properties. The information also includes standardized technical functions in accordance with IEC 61360 [10] and ISO 13584-42:2010 [11]. Generic information of assets is available in the IEC Common Data Dictionary (IEC CDD). The shown layer architecture is structured into six aspects necessary to describe the properties of an asset in the information world. An example is shown here to describe a temperature sensor in accordance with RAMI4.0: – The asset layer represents the physical environment of an asset, for example a temperature transmitter measuring a physical temperature (object in the physical world). NOTE Simulation is a representation of the physical world in the integration layer. – The physical information "temperature" is to be converted into a (binary) number in a specific format. Thus, the integration layer is the conversion layer from the physical world into the information world. – The (binary) value can be transferred to other assets too. For this purpose, the task is done by the communication layer. – Because an object always has a purpose, it has a technical function which is located in the functional layer accessing. It also had managing information that is reflected in the information layer. – Finally, the business layer reflects the fact that every object has financial, legal and delivery aspects like cost, regulation, contracts, and delivery time. They are expressed in the information world through standardized variables with a specific format. – Not all layers are filled in with information. One or more layers can also be empty and skipped.
	Examples	-
	Rationale for the pattern	Extend the IoT reference architecture to support smart manufacturing
	Guidance	Pattern guidance: The pattern is used to characterize any object by function, time, and allocation to someone or something. The purpose is to mirror that information into the information world by coding it in accordance with IEC 61360 [10] and ISO 13584-42:2010 [11]. Standardized properties can be found in IEC 61360 [10] and with a wider scope in ECLASS [12].

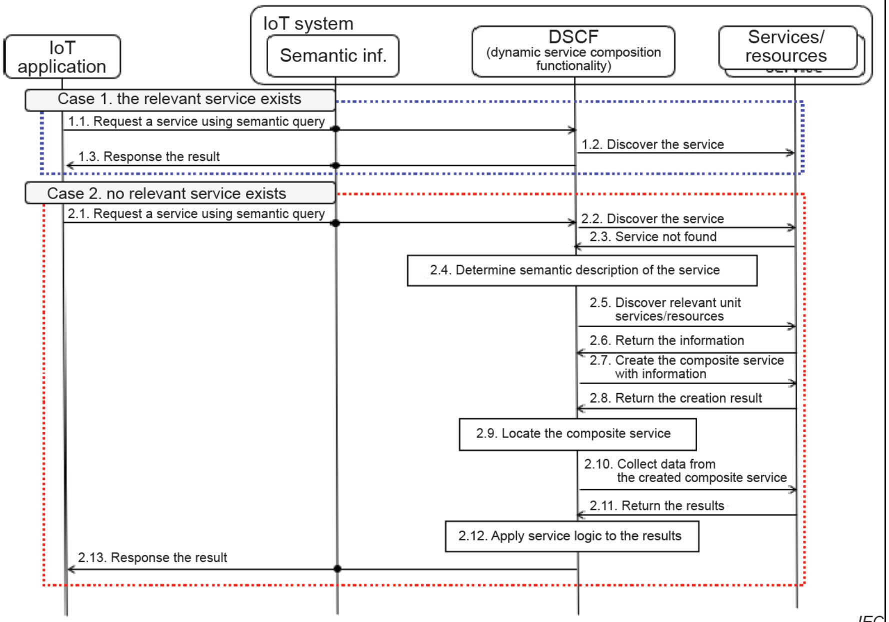
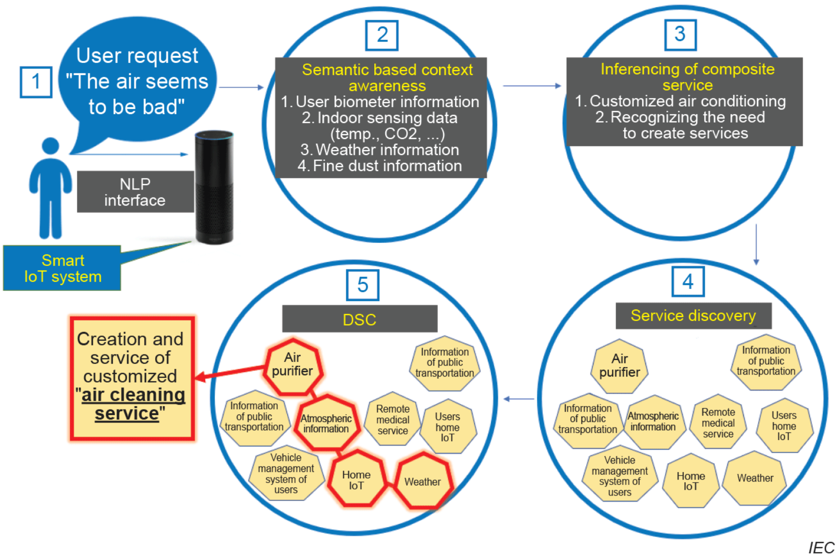
A.3 Dynamic IoT system pattern

The dynamic IoT system pattern is described in Table A.2.

Table A.2 – Dynamic IoT system pattern

Information	Name	Dynamic IoT system pattern
	Related patterns	-
Problem		Dynamic service composition is based on the capabilities that can be provided by an IoT device. Dynamic service composition can dynamically combine the capabilities of multiple components into one IoT system. Dynamic service composition is created by cooperation between IoT devices using artificial intelligence, context-awareness, and service profile.
Known context	Specific context	
	Related context	

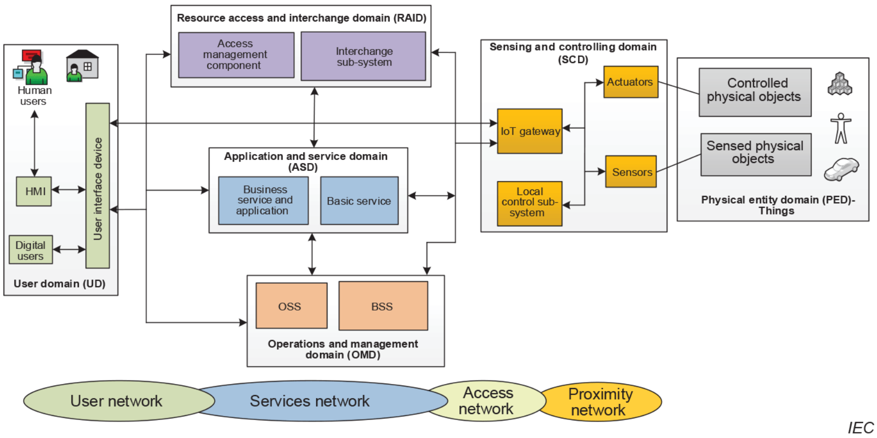
Solution	Architecture models	DSC components:  IoT application Semantic query IoT system Virtual entity [Composite service] Composite attribute #1 Composite attribute #2 Attribute #1 Attribute #2 Attribute #3 Attribute #4 Dynamic service composition (DSC) functionality Abstract entity Unit service #1 Unit service #2 Unit service #3 Unit service #4 Unit service #5 ... Unit service #n Physical entity Unit service (1) Unit service (2) Unit service (3) Unit service (4) Unit service (2) Unit service (5) Unit service (6) Unit service (5) Unit service (7) Location #1 Location #2 Location #n IEC **Figure A.2 – DSC components** Figure A.2 describes: – Physical entity: It describes the things, places, and the services in the real-world. – Abstract entity: The abstracted physical things that are recognized in the IoT system. – Virtual entity: It is the IoT service or resource that is created by combining multiple abstract things. – Composite service: New virtual services provided through a combination of unit services. – Attribute: Individual attributes of the information provided by the composite service. – Composite attribute: It is the attributes that are created by combining individual attributes. Figure A.3 describes a sequence diagram for DSC. Figure A.4 describes an example of DSC.

		Sequence diagram for DSC:  IoT application IoT system Semantic inf. DSCF (dynamic service composition functionality) Services/ resources Case 1. the relevant service exists 1.1. Request a service using semantic query 1.2. Discover the service 1.3. Response the result Case 2. no relevant service exists 2.1. Request a service using semantic query 2.2. Discover the service 2.3. Service not found 2.4. Determine semantic description of the service 2.5. Discover relevant unit services/resources 2.6. Return the information 2.7. Create the composite service with information 2.8. Return the creation result 2.9. Locate the composite service 2.10. Collect data from the created composite service 2.11. Return the results 2.12. Apply service logic to the results 2.13. Response the result IEC **Figure A.3 – Message flow in DSC**
	Examples	Example of DSC:  1 User request "The air seems to be bad" NLP interface Smart IoT system 2 Semantic based context awareness 1. User biometer information 2. Indoor sensing data (temp., CO2, ...) 3. Weather information 4. Fine dust information 3 Inferencing of composite service 1. Customized air conditioning 2. Recognizing the need to create services 4 Service discovery Air purifier Information of public transportation Atmospheric information Remote medical service Users home IoT Vehicle management system of users Home IoT Weather 5 DSC Creation and service of customized "air cleaning service" IEC NLP natural language processing **Figure A.4 – Home smart air cleaning service**
	Rationale for the pattern	Benefits: DSC is convergence unit service. DSC dynamically connects different services through mutual cooperation between neighbouring devices that are based on unit services that can be provided by IoT devices. DSC is similar to CPS. However, CPS tends to focus on traditional static system design.
	Guidance	

A.4 IoT enterprise system pattern

The IoT enterprise system pattern is described in Table A.3.

Table A.3 – IoT enterprise system pattern

Information	Name	IoT enterprise system pattern
	Related patterns	The IoT enterprise system pattern is using the IoT enterprise taxonomy pattern
Problem		
Known Context	Specific context	
	Related context	
Solution	Architecture models	The system deployment model describes the generic components including IoT devices, subsystems, and networks to form an IoT system. While the functional view describes an IoT system through its functional components, the system deployment model kind describes it through its implemented components. The IoT component capability model can be used to describe the system deployment. The system deployment view describes the following aspects: – key components (subsystems, devices, networks) of an IoT system along with a technical description of the components and their capabilities; – the relationship between components, including the structure of an IoT system, the distribution of components, and the topology of the interconnectivity of the components.  Resource access and interchange domain (RAID) Access management component Interchange sub-system Human users HMI Digital users User interface device User domain (UD) Application and service domain (ASD) Business service and application Basic service OSS BSS Operations and management domain (OMD) Sensing and controlling domain (SCD) Actuators IoT gateway Sensors Local control sub-system Controlled physical objects Sensed physical objects Physical entity domain (PED)-Things User network Services network Access network Proximity network IEC **Figure A.5 – Example system deployment model** In Figure A.5, enterprise system deployment is shown together with all the entities that are involved in each domain and the connections between them. The entities in each domain are general and optional, depending on specific applications. There are four different kinds of networks to connect the physical components in the six domains of an IoT system: proximity network, access network, services network, and user network. Physical systems and sub-systems in the physical entity domain (PED): The PED mainly consists of sensed physical objects and controlled physical objects, which are related to IoT applications and are of interest to users. A sensed physical object is a physical entity from which information is acquired by sensors. A controlled physical object is a physical entity which is subject to actions of actuators. Systems/sub-systems in the sensing and controlling domain (SCD): In the SCD, the entities consist primarily of sensors, actuators, and IoT gateways. Sensors sense properties of physical entities while actuators change properties of physical entities. Sensors acquire information about a property of a physical entity (physical, chemical, biological properties). Actuators change properties of entities. Both sensors and actuators can interact with physical entities independently or collaboratively.

		IoT gateways are devices which connect SCD with other domains. IoT gateways provide functions such as protocol conversion, address mapping, data processing, information fusion, certification, and equipment management. IoT gateways can be either independent equipment or integrated with other sensing and controlling devices. The IoT gateway can also perform security functions for constrained IoT devices using the gateway for connectivity to networks. The SCD might also contain local control systems which are used to run control services. That is, components for local management of IoT gateway capabilities in scenarios where the IoT gateway is expected to work with or without upstream connectivity. Systems/sub-systems in the application and service domain (ASD): The purpose of the ASD subsystem is to host the core functions. Core functions are services and applications that deliver the IoT system functionality to the users (human or digital). The ASD subsystem provides basic services. Basic services include computing services such as data access, data processing, data fusion, data storage, identity resolution. Basic services also cover geographic information service, user management, and inventory management. The ASD subsystem also hosts business services and applications that are built on the generic services. The ability to host applications is one of the services provided by the IoT systems. Systems/sub-systems in the operation and management domain (OMD): The OMD subsystem hosts components responsible for management of IoT devices and control of the operation of the IoT system. The purpose is to guarantee that the equipment and systems operate safely and reliably. Also, it monitors the system to ensure that relevant laws and regulations are not violated. The OMD contains the operational support system (OSS) and the business support system (BSS). The OSS is responsible for handling the overall operation of the IoT system. The OSS includes capabilities for monitoring and managing all entities of the IoT system over their complete life cycle. The OSS includes compliance systems which enable checking of the IoT system for compliance with laws, regulations, and enterprise policies. A business support system (BSS) is responsible for realization of the business aspects of the IoT system. The business functions include customer relationship management (CRM), subscription management, billing, and payment processing. Systems/sub-systems in the user D^domain (UD): The user domain contains both human users and digital users. Digital users are devices of some type, and they interact directly with other entities in the IoT system by network interfaces or application programming interfaces. Human users interact using a user device which contains some form of HMI. HMI subsystem contains the devices and supporting software that allow human users to interact with the IoT system. Depending on user role, different aspects of the system are presented for observation and control. Systems/sub-systems in the resource access and interchange domain (RAID): The RAID contains access management component and interchange subsystem. Access management component authenticates and authorizes external users of the IoT system wanting to access the capabilities of the IoT system. Reverse access management is also necessary, when the IoT system leverages information and capabilities that are provided by a partner IoT system. Interchange subsystem provides exposure of capabilities within the IoT system. Such capabilities include applications, data, and services in the ASD as well as administration and business capabilities in the OMD. The latter provide the basis for automation of setting up the trust relationships that provide the authorization data.
	Examples	
	Rationale for the pattern	
	Guidance	

A.5 IoT enterprise networking pattern

The IoT enterprise networking system pattern is described in Table A.4.

Table A.4 – IoT enterprise networking system

<table>
<tr><td rowspan="2">Information</td><td>Name</td><td>IoT enterprise networking pattern</td></tr>
<tr><td>Related patterns</td><td>The IoT enterprise networking pattern is using the IoT enterprise system pattern</td></tr>
<tr><td colspan="2">Problem</td><td></td></tr>
<tr><td rowspan="2">Known context</td><td>Specific context</td><td></td></tr>
<tr><td>Related context</td><td></td></tr>
<tr><td>Solution</td><td>Architecture models</td><td>
Network systems overview:

This section describes the principal communications networks which are involved in enterprise IoT systems and the entities with which they connect. The four principal communications networks are shown in Figure A.6.

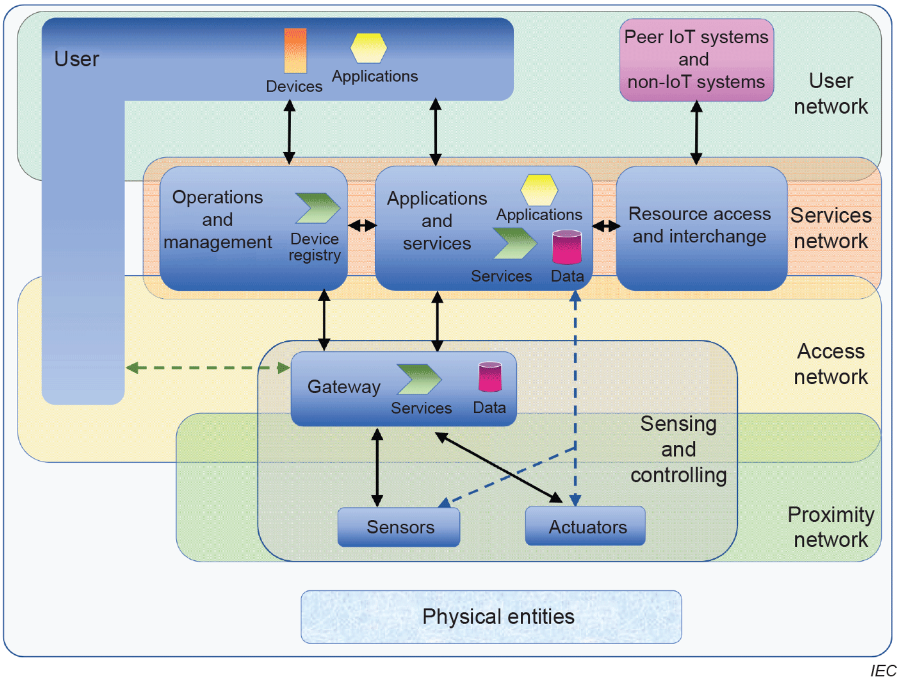


Figure A.6 – Networking model

Interconnected networks provide communication connectivity, including data links. Data links can be point-to-point links in or between IoT systems, both interdomain and intradomain, and with other systems and organizations. The connected networks should maintain connectivity from one network to another. The key role of the networks is to support and provide communication and data exchange activities and interactions. The types of the activities and interactions between two entities, between two domains, or between two IoT systems determine their relationships between the entities, domains, and IoT systems. Although the interdomain communication networks are not designated as part of one of the six domains, these networks play a critical role in an IoT system. Depending on the infrastructure of IoT systems, the interdomain communication networks can be local area network, Internet, intranet, enterprise backbone network, or wide area network. Business-to-business (B2B) networks are also considered as interdomain communication networks.
</td></tr>
</table>

		Proximity network: This network exists within the SCD. Its main task is to connect sensors and actuators to the IoT system. Proximity networks are typically local and limited in range. Proximity networks are necessary because sensors and actuators are low power or are in locations that make wide area connections (such as the Internet) difficult or impossible to provide. Proximity networks can use specialized protocols instead of generic protocols such as IP. It is possible that individual sensors and actuators have limited power and limited hardware capabilities., Because of this, simple, local, and low-power networks are needed to connect them to gateways. Gateways are more powerful and can in turn connect to access networks. Examples of proximity networks include IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN), ZigBee, and Wireless HART. Proximity networks can involve the use of an address translation capability to translate between their local addressing schemes and addressing schemes that are used on access networks. Access network: Access networks are typically wide area networks connecting devices in the SCD to the other domains – the ASD and the OMD. Access networks typically connect to gateways, but when sensors and actuators are more capable and with a limited connection situation, they can connect directly to access networks (dashed lines in Figure A.6). A range of technologies can be used in access networks. The technologies include wired connections (broadband, ADSL or fibre) and wireless connections including wireless LANs, and mobile (cellular) networks. The technologies also include low-power wide area networks, and satellite links (particularly for remote locations). Access networks typically use IP. Access networks can involve the use of a device registry. A device registry holds data about the IoT devices that are associated with the IoT system and how to communicate with them. Services network: The services network connects the applications and services in the ASD, the RAID and the OMD. Services networks are typically wired networks within data centres, running IP-based protocols. Services networks can include both Internet elements and also (private) intranet elements. It is typical for intranet networks to be used where the elements of the other domains exist within a single data centre. Where communication spans multiple data centres, various network technologies can be used, including both dedicated connections and Internet connections. User network: This network connects the user domain with the ASD and OMD. It also connects peer IoT systems and non-IoT systems with the RAID. This network is typically based on public Internet elements and uses IP. Such networks can use any of the technologies that are commonly used to carry Internet traffic, including both wired and wireless systems. Implementation of communication networks: Each of the principal communications networks can be implemented by a range of different network technologies. Which technology to use is dependent on the characteristics and requirements of the IoT system. IoT system implementations can use multiple instances of each of these networks to create complete solutions. The key to the interoperability among IoT systems is how data and information is correctly transferred from one type of network to another type of network. One communication network component that takes a vital role to glue the dissimilar communication networks is the gateways that are designed for IoT, that is IoT gateways. IoT communities are adapting the emerging communication networks that are designed for IoT applications: Still, accommodating legacy communication networks is also needed to promote the interoperability in IoT applications. Network topology is also an important aspect of the IoT systems. The potential IoT systems can use different network topologies to successfully support IoT functionality and capability. The representative network topologies are, for example, point-to-point (permanent or switched), bus (linear or distributed), star (extended or distributed). Network topologies can also be ring, mesh (fully connected or partially connected), ad hoc, and hybrid (combination of two or more of the topologies above). In Figure A.6, the user domain is shown spanning both the user network and the access network. Figure A.6 describes cases where user devices and their applications connect directly to the SCD. An example is when the user device is a smartphone which contains sensors.

	Examples	
	Rationale for the pattern	
	Guidance	
NOTE Any tradenames and trademarks in this table are examples of suitable products available commercially. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of these products.		

A.6 IoT enterprise usage pattern

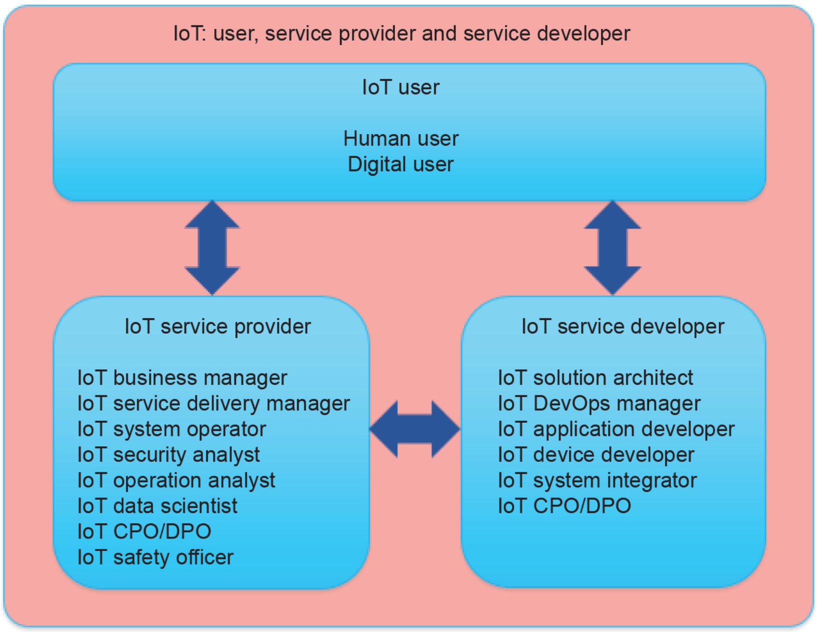
The IoT enterprise usage pattern is described in Table A.5.

Table A.5 – IoT enterprise usage pattern

Information	Name	IoT enterprise usage pattern
	Related patterns	-

Problem		The functional view shows the necessary functions and dependencies of the IoT system. The functional view also shows how the IoT system is developed, tested, operated, and used from a user perspective. This pattern addresses the following concepts:
		– activities;
		– roles and subroles;
		– services and cross-cutting aspects.

Known context	Specific context	
	Related context	
Solution	Architecture models	Description of the roles, subroles, and related activities: All IoT related activities can be categorized into three user groups as listed below: 1) IoT service provider. 2) IoT service developer. 3) IoT user. Figure A.7 gives an overview of the three user groups (roles) and their subroles. Blue arrows show their interaction when the system is in use. Details about the roles and subroles are described below.



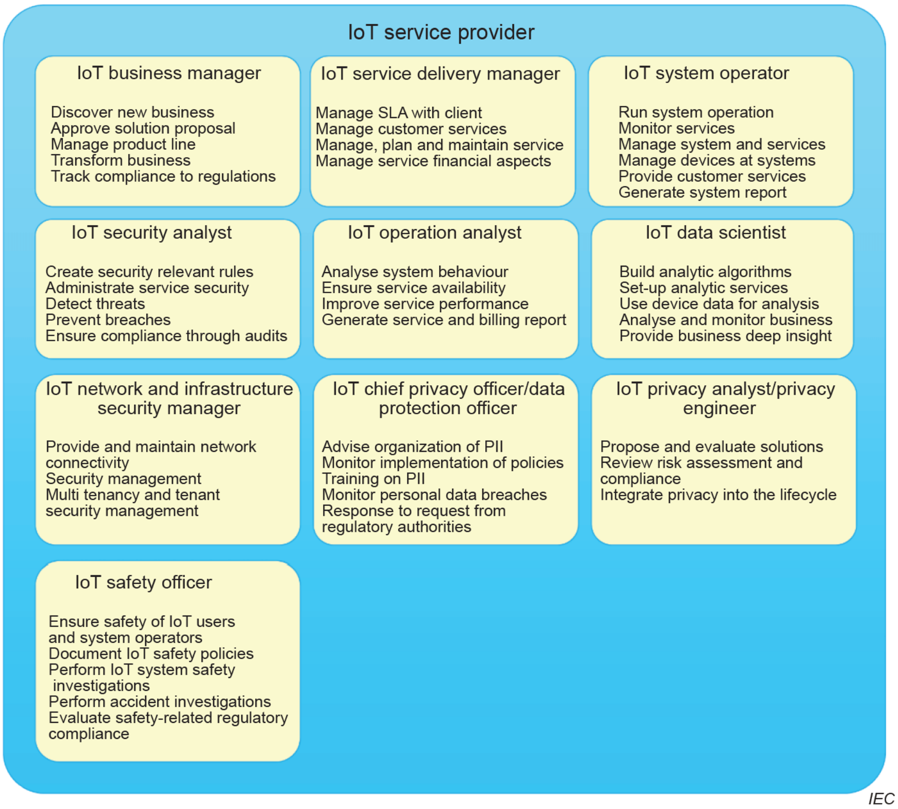
IEC

Figure A.7 – Roles present when the system is in use

IoT service provider role:

The role of service provider is to manage and to operate IoT services. IoT service providers can also provide network connectivity. Security of this connectivity must be addressed and maintained. Also, in cloud based IoT services, security management, multitenancy, tenant security, and separation must be managed at different layers. From the hardware up the stack to the application layers. These services are dependent on the type of service that is offered from the data centre (SaaS, PaaS, IaaS).

Figure A.8 shows the activities which relate to the subroles of IoT service provider.


IoT service provider
IoT business manager
Discover new business
Approve solution proposal
Manage product line
Transform business
Track compliance to regulations
IoT service delivery manager
Manage SLA with client
Manage customer services
Manage, plan and maintain service
Manage service financial aspects
IoT system operator
Run system operation
Monitor services
Manage system and services
Manage devices at systems
Provide customer services
Generate system report
IoT security analyst
Create security relevant rules
Administrate service security
Detect threats
Prevent breaches
Ensure compliance through audits
IoT operation analyst
Analyse system behaviour
Ensure service availability
Improve service performance
Generate service and billing report
IoT data scientist
Build analytic algorithms
Set-up analytic services
Use device data for analysis
Analyse and monitor business
Provide business deep insight
IoT network and infrastructure security manager
Provide and maintain network connectivity
Security management
Multi tenancy and tenant security management
IoT chief privacy officer/data protection officer
Advise organization of PII
Monitor implementation of policies
Training on PII
Monitor personal data breaches
Response to request from regulatory authorities
IoT privacy analyst/privacy engineer
Propose and evaluate solutions
Review risk assessment and compliance
Integrate privacy into the lifecycle
IoT safety officer
Ensure safety of IoT users and system operators
Document IoT safety policies
Perform IoT system safety investigations
Perform accident investigations
Evaluate safety-related regulatory compliance

IEC

Figure A.8 – IoT service provider subroles and activities

IoT service developer:

The roles of the IoT service developer include implementation, testing, and integration of IoT services with the IoT platform. Subroles of the IoT service developer are described as follows.

An IoT solution architect proposes, proves, and deploys the IoT enabled platform to the LoB. An IoT solution architect also decides on integration strategies and architectures for the new IoT enabled platform, existing business systems, and devices in production.

An IoT development operations manager sets up, configures, and operates the IoT enabled platform, relevant services. The IoT development operations manager also acts as a project manager by supporting IT services for LoB operations and development.

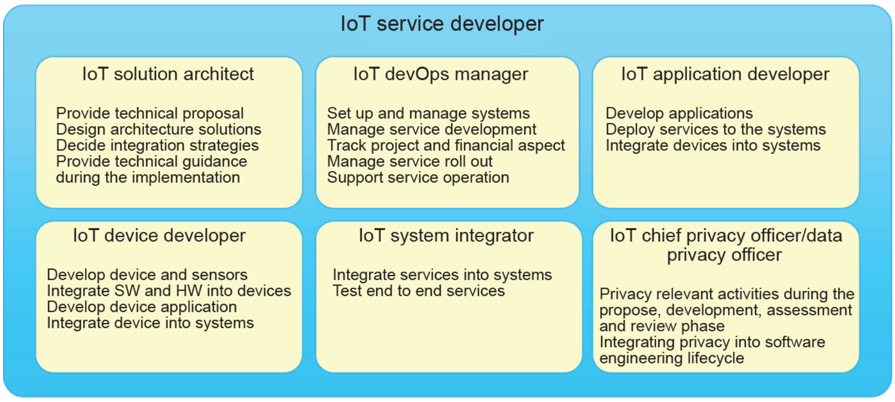
An IoT application developer works in the LoB, in IT or with a third party, developing IoT industry applications for the LoB. The IoT application developer uses development operation capabilities to develop, deploy, and fix applications that integrate IoT devices, data, and services.

An IoT device developer integrates hardware and software into devices and applications, developing and maintaining device firmware that securely connects devices to an IoT-enabled platform.

An IoT system integrator tests and integrates IoT services with the IoT enabled platform.

An IoT CPO or DPO has several duties like designing cutting-edge products and services that leverage big data while preserving privacy. The IoT CPO or DPO is also proposing and evaluating solutions (privacy-enhancing technologies) to mitigate privacy risks, conducting privacy-related risk assessments and compliance reviews. Furthermore, the IoT CPO or DPO is responding to incidents and integrating privacy into the software engineering life cycle phases.

All IoT service developer subroles and their activities are shown in Figure A.9.


IoT service developer
IoT solution architect
Provide technical proposal
Design architecture solutions
Decide integration strategies
Provide technical guidance during the implementation
IoT devOps manager
Set up and manage systems
Manage service development
Track project and financial aspect
Manage service roll out
Support service operation
IoT application developer
Develop applications
Deploy services to the systems
Integrate devices into systems
IoT device developer
Develop device and sensors
Integrate SW and HW into devices
Develop device application
Integrate device into systems
IoT system integrator
Integrate services into systems
Test end to end services
IoT chief privacy officer/data privacy officer
Privacy relevant activities during the propose, development, assessment and review phase
Integrating privacy into software engineering lifecycle

IEC

Figure A.9 – IoT service developer subroles and activities

IoT user:

The IoT user is the end user of IoT services and can be categorized into human users and digital users.

Human users are individuals who use IoT services. Digital users are nonhuman users of the IoT system; they can include automation services that act on behalf of a human user.

All IoT user subroles and their activities are shown in Figure A.10.

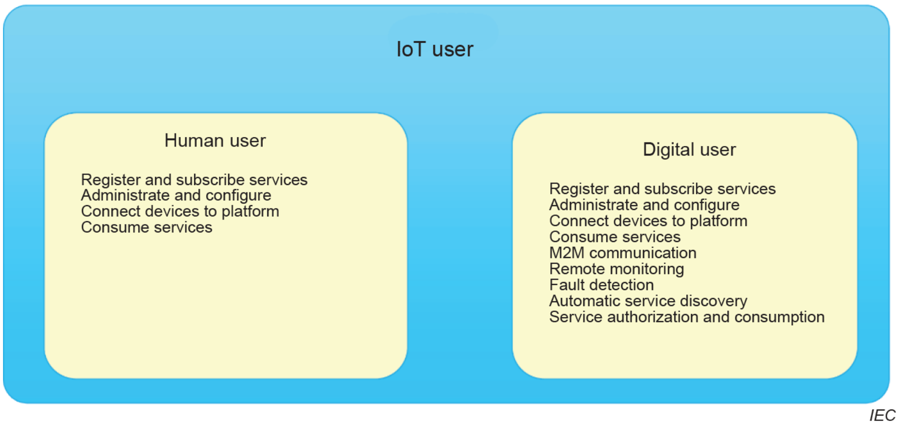

IoT user
Human user
Register and subscribe services
Administrate and configure
Connect devices to platform
Consume services
Digital user
Register and subscribe services
Administrate and configure
Connect devices to platform
Consume services
M2M communication
Remote monitoring
Fault detection
Automatic service discovery
Service authorization and consumption
IEC

Figure A.10 – IoT subroles and activities

Mapping activities, roles and IoT systems in domains:

The usage view addresses the concerns of expected system usage.

Roles and activities involving IoT users to deliver functionality achievable with the fundamental system capabilities are represented by this view. Activities which create, implement, test, integrate, and operate IoT services in wanted systems can require interaction among individuals with different roles or skills (see Figure A.8).

Table A.6 and Table A.7 provide an overview of activities and their relevant roles.

Figure A.11 and Figure A.12 show some examples of using IoT systems from different activity perspectives.

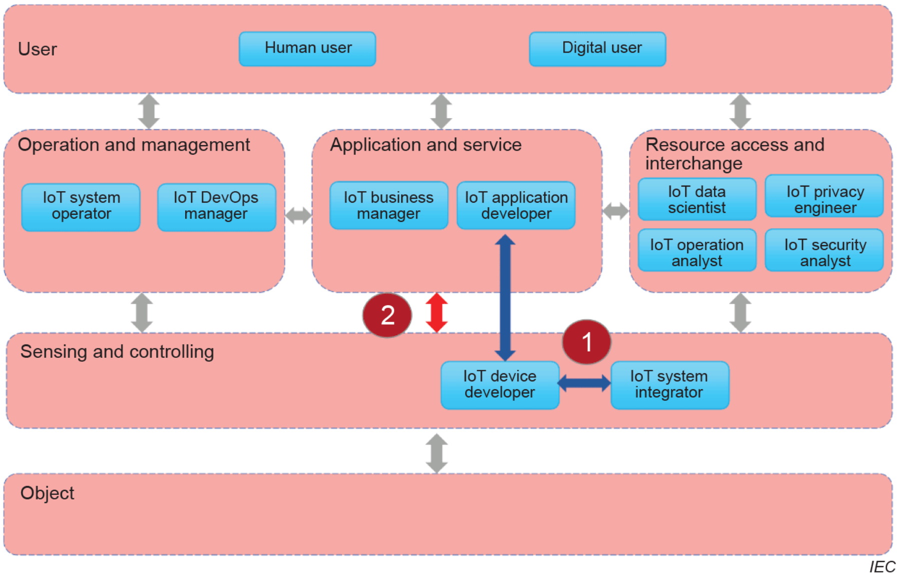


Figure A.11 – Activities of device and application development

Figure A.11 shows an example of activities and information exchange during device application development between device developers, system integrators, and application developers. An example of a specific user activity is connecting a new device to the IoT platform. The boxes in Figure A.11 represent the human users (in this case developers and operators) of IoT systems. The six domains of an IoT system are represented by boxes with dashed lines. For this activity:

– The device developer communicates with the system integrator during the implementation phase. They discuss API definitions and functional behaviour between the device and the IoT platform and agree to a specification.

– The application and device developers implement and test APIs and their functions that are related to the device and the IoT platform. At this stage, devices in the SCD are connected to IoT systems in the ASD and end-to-end functions can be tested.

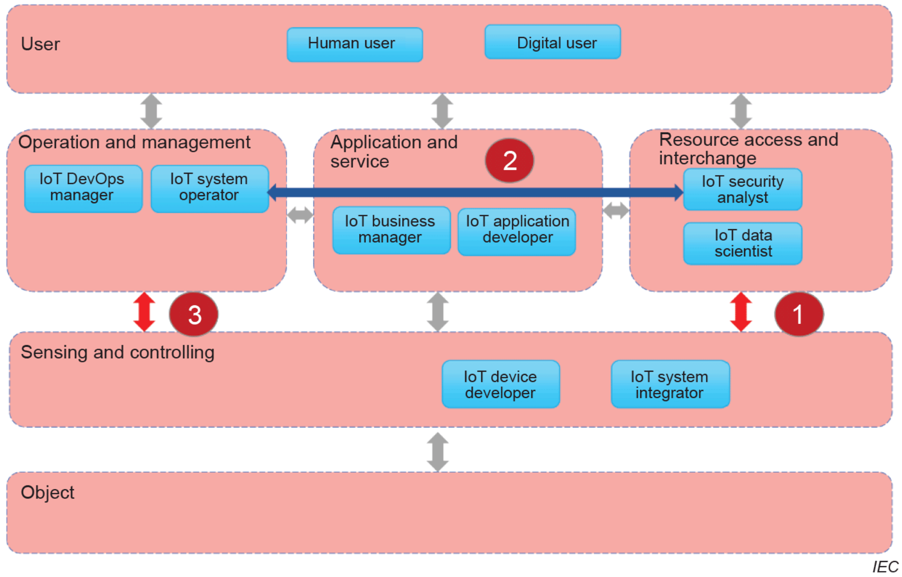
		 User Human user Digital user Operation and management IoT DevOps manager IoT system operator Application and service 2 IoT business manager IoT application developer Resource access and interchange IoT security analyst IoT data scientist 3 1 Sensing and controlling IoT device developer IoT system integrator Object IEC **Figure A.12 – Using device data for security-related analytics and operations.** Figure A.12 shows an example of activities that are involved in using device data for security-related analytics and operations. In this case, the users of the IoT systems are the data analyst and security operator. Activities include, but are not limited to: – When the device is configured and connected to the communications system, usage data can be sent to the IoT systems in the RAID. The security analysts and data scientists can use the collected device usage data to perform security-related analyses. – Security analysts communicate with system operators with findings and results from their analyses. – Security analysts together with system operators proactively create rules to protect systems and to prevent breaches.
	Examples	
	Rationale for the pattern	
	Guidance	

Table A.6 – Overview of activities and roles

Activities	Roles	IoT systems in domains
Use device data for analytics.	IoT data scientist, IoT security analyst, IoT operation analyst	Operation and management domain, access and communication domain
Use real-time, historic, and big data for applications and analytics.	IoT data scientist, IoT operation analyst, IoT security analyst, IoT service delivery manager	Application and service domain, operation and management domain, sensing and controlling domain, resource access and interchange domain
Make and operate analytics to run business.	IoT data scientist, IoT operation analyst, IoT application developer, IoT DevOps manager	Application and service domain, resource access and interchange domain

Table A.7 – Overview of enterprise activities and roles

Activities	Roles	IoT systems in domains
Device and application development.	IoT DevOps manager, IoT device developer, IoT application developer	Application and service domain, sensing and controlling domain
Operation of devices, connectivity, and applications.	IoT system operator, IoT service delivery manager	Operation and management domain, application and service domain
Integrate, operate, and control data stores and business.	IoT solution architect, IoT DevOps manager, IoT system operator, IoT system integrator, IoT service delivery manager	Application and service domain, operation and management domain
Bring in analytics to dashboard.	IoT DevOps manager, IoT data scientist, IoT application developer	Application and service domain, operation and management domain, resource access and interchange domain
Monitor system state, act on security risks and breaches.	IoT system operator, IoT security analyst	Operation and management domain
Track compliance with regulations.	IoT business manager, IoT security analyst	Application and service domain, user domain

Annex B
(informative)

Guidance on the use of ISO/IEC/IEEE 42010:2022

B.1 Overview

The purpose of this Annex B is to provide information about how to conform to ISO/IEC/IEEE 42010:2022 requirements and recommendations when developing an architecture description (AD).

The distinction between an architecture, which is the abstract conceptualization, and a description of that architecture, which is an AD, is often conflated in discussion and casual writing.

As specified in ISO/IEC/IEEE 42010:2022, all systems have an architecture (an abstract human conceptualization of the system's structure, function and fitness-for-purpose exhibited in various ways), and this abstract conceptualization is expressed as an AD.

ISO/IEC/IEEE 42010:2022 provides terms, definitions, and relationships for expressing architecture elements and relationships among those elements suitable for the creation and use of an architecture description.

B.2 Systems and architectures

The expression of an architecture that is suitable for communication to different stakeholders is called the architecture description (AD). This document (ISO/IEC 30141) is an AD prepared for the purpose of communicating and guiding the design and development of IoT systems. It serves as a reference architecture description that is applicable to a set of IoT systems of interest. In that capacity, this document is known as the "reference architecture for the Internet of Things".

Since there can be multiple architectures of a system with different terms and interpretations, architects document a particular architecture as an AD for the purpose of communicating information about that architecture to others. Several architects can conceptualize architectures for the same system differently because they are communicating different information, often to different stakeholders.

B.3 Elements in ISO/IEC/IEEE 42010:2022 used in ISO/IEC 30141 IoT reference architecture description

B.3.1 Overview

ISO/IEC/IEEE 42010:2022, 5.2 presents the key concepts and relationships for ADs. Figure B.1 provides a partial schematic representation of those objects and relationships.

The brief explanation of the key concepts of an AD below are adapted from ISO/IEC/IEEE 42010:2022, 5.2 with additional explanatory text.

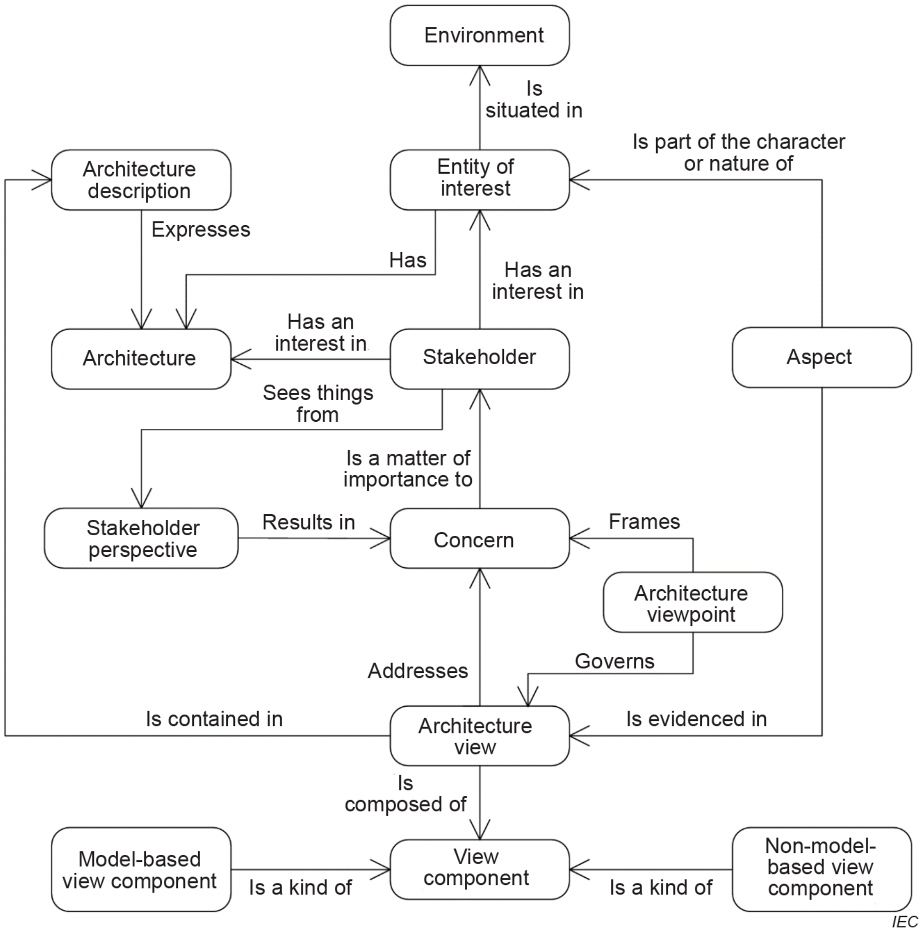
B.3.2 Stakeholders, perspective, and concerns

Stakeholders are parties with direct or indirect interests in the set of systems of interest. The primary stakeholder of ISO/IEC 30141 is the system architect.

Concerns are matters of interest or importance to one or more stakeholders. In the case of ISO/IEC 30141 the system architect's concerns are primarily about the structure or performance of the IoT systems.

During the life cycle of the system of interest, concerns arise at any time. However, because a system's architecture usually endures for the entire life cycle, only concerns of architectural significance that impact the structure, behaviour or fitness-for-purpose of the system are pertinent to the architecture. Concerns can manifest in various ways in relation to stakeholder's needs, architecture goals, expectations, responsibilities, requirements, design constraints and assumptions. Concerns can also manifest in recognition of dependencies, quality attributes, architecture decisions, risks, or other issues.

Concerns can pertain to influences exerted upon or by a system of interest, including developmental, technological, business, operational, organizational, political, economic, legal, regulatory, ecological, social, and physical influences. Concerns can also pertain to design influences such as internal structural features and component interoperability, particularly when architecting a system of systems or an enterprise.



[Source: ISO/IEC/IEEE 42010:2022, Figure A.1]

Figure B.1 – Conceptual model of an architecture description

B.4 Viewpoints, model kinds, legends, correspondences, and correspondence methods

Given a collection of concerns, they are categorized as manageable groups so that they are addressed in the architecture. Each group of one or more related concerns becomes a basis for an architecture viewpoint relative to that group of concerns. In other words, architecture viewpoint frames a group of one or more related concerns. Architecture viewpoints establish the conventions to understand, create and use architecture views.

The viewpoint identifies model kinds and legends which govern the corresponding architecture views. A model kind determines the conventions for model-based views. A legend documents the conventions for non-model-based views. These conventions include the intended uses, the terminology, the notations and their syntax and semantics and symbology of its governed models. A model kind or legend can be used by more than one viewpoint in an architecture description. The viewpoint identifies correspondences in the architecture description. Correspondences are an identified or named relation between two or more architecture description elements.

The viewpoint identifies correspondence methods. Correspondence methods capture intended relationships that are to be enforced on correspondences within and between architecture description elements.

The important thing to remember about viewpoints is they provide the conventions for defining or understanding the content of an architecture description. Architecture viewpoints do not directly define anything about the architecture, only how to describe parts of the architecture description.

The legends are contained within the viewpoints and explain how to interpret each view component. They are used to formally define what is present in any instance of a model that satisfies one or more concerns.

B.5 Views and models

Architecture models and architecture views are governed by the conventions provided by architecture viewpoints. The resultant artefacts address the concerns framed by the viewpoint. The number of models and number of views necessary to address the concerns framed by a particular viewpoint depend upon many factors related to the extent the concerns are addressed and the viewpoint specification itself.

B.6 Correspondences

As specified in ISO/IEC/IEEE 42010:2022, correspondences are an identified or named relation between two or more architecture description elements. In this document, correspondences are also used to describe the relationships between the elements of an architecture description of the RA and the elements of architecture descriptions of other reference architecture standards and other standards that can be used jointly.

EXAMPLE Consider a new RA standard that is used jointly with this document. The new RA includes a clause describing the correspondences between the architecture descriptions of the two standards (the new RA standard to be developed and this document).

Annex C
(informative)

Characteristics for IoT systems in particular contexts

C.1 Common characteristics

C.1.1 Legacy support

Legacy support makes it possible to continue to use older components, even where these components embody technologies that are no longer standard or approved. That applies to services, protocols, devices, systems, components, technologies and similar.

Support of legacy component integration and migration can be important. It is also important to ensure that the design of new components and systems does not unnecessarily limit future system evolution. To prevent prematurely stranding legacy investment, a plan for adaptation and migration of legacy systems is important. Care should be taken when integrating legacy components to ensure that security and other essential performance and functional requirements are met. Legacy components can increase risk and vulnerabilities. Since current technology becomes legacy technology in the future, it is important to have a process in place for managing legacy aspects of any given system. Legacy support is crucial for IoT, where different connected devices can have very different life cycles and update schedules, often in concert with the life cycle of the physical and information systems with which they are integrated.

C.1.2 Network connectivity

Network connectivity is a core concept of IoT. IoT systems rely on the ability to exchange information in a structured manner through many kinds of networks.

The choice of what network connectivity capabilities to implement is often crucial for the IoT system and is typically carefully considered. New communication protocols are continuously developed and made available. The connectivity choices are driven by the specific applications and use cases.

The expected volume of the transmitted data will also be a base for the decision about the choice of network communication. Communication networks often differ a lot in regard to the cost for data volumes. To some extent this can be encountered by using edge technology to aggregate or in other ways reduce the amount of data transferred through the network.

C.1.3 Unique identification

It is essential that the entities in an IoT system can be distinguished from each other. This enables interoperability and global services across heterogeneous IoT systems. It is important for entities to be uniquely identifiable within a given context so that IoT systems can appropriately monitor and communicate with specific entities. Some devices can be hidden behind IoT gateways, or information consolidated to protect privacy (see ISO/IEC 15045-1:2004 [13]). A variety of identification schemes can be supported in specific implementations of IoT systems to meet the application requirements.

NOTE IoT components are often registered in an IoT system at the same time as they are placed at a physical location. Because of this, it might be tempting for a system owner to use the physical location of the IoT component as identifier. However, doing so might cause difficulties when moving IoT components. Also, more IoT components can be deployed at the same spot over time. In that case, differentiation between the component might cause difficulties if geographical location is used as base for identification. It is important to consider how to replace IoT components, like faulty or outdated components. So, it is important to preserve metadata and other relevant information related to the component.

C.1.4 Well-defined components

IoT entities are considered to be well-defined when an accurate description of their capabilities and characteristics is available, including any associated uncertainties.

Using or producing "not so well defined" components might jeopardize an entire IoT system. For example, if the interface of a component is not described in detail, it might be impossible to protect the device from cyberattacks or even judge its trustworthiness. If the behaviour of a component is not known, it is difficult to replace with another component. Furthermore, such components can be difficult to debug. Even if it might be tempting to use "not so well defined" components (that can sometimes be cheaper), the risk should be considered.

C.1.5 Auto-configuration

Auto-configuration is the automatic configuration of devices based on predefined rules. Auto-configuration includes automatic networking, automatic service provisioning and plug and play.

Auto-configuration is useful for large-scale IoT systems whose configurations change dynamically over time. Promotion of faulty component elimination and timely maintenance by auto-configuration greatly benefits users with demanding reliability requirements. The system owner might want to compare the costs for auto-configuration or manual work. That is, having auto-configuration parts of the IoT system compared to the manual work required to set up and maintain the IoT system without auto-configuration for those parts.

C.1.6 Content-awareness

Content-awareness is knowledge about the information in an IoT component and its associated metadata. Devices and services with content-awareness can adapt interfaces, abstract application data, improve information retrieval precision, discover services, and enable appropriate user interactions.

It is important to decide the level of content-awareness for the IoT system and IoT components. As with auto-configuration, there is most likely a balance between the cost for such capabilities and the needs. The needs for content-awareness often relate to the compliance needs for the data that are treated by the IoT system.

C.1.7 Context-awareness

Context-awareness is accomplished by metadata of an IoT device, service, or system, describing the context in which the IoT device operates. This can be information such as when (time awareness), where (location awareness), or in what order (awareness of sequence of events) one or more observations occurred in the physical world.

Typically, it is a good idea to investigate all parts of the IoT system to determine where context-awareness is needed and how it can be achieved. At IoT device level, a good start is to ensure that the IoT device can deliver all meta-information that is needed for the wanted context-awareness.

C.1.8 Discoverability

Discoverability is about how an endpoint on the network can be found dynamically and report its services and their capabilities through a query mechanism or self-advertising mechanism. The endpoints can be IoT devices, services, and applications, or even users.

The wanted level of discoverability is often decided by the business case for the IoT system. Services that are connected with an IoT system can indicate what information can be found by a discovery or lookup service in accordance with predefined rules for each market segment. Discovery and lookup services allow IoT systems to locate other devices, services, or systems. These can be located based on parameters such as geographical location, capabilities, interfaces, accessibility, ownership, security policy, operational configuration, or other relevant factors.

C.1.9 Manageability

Manageability includes all parts of the IoT system and IoT environment, like device management, network management, system management, interface maintenance, rules, and alerts. In an IoT environment, many IoT devices, networks, and systems operate autonomously. Often large scale and geographic span of IoT systems places demands on the possibility to manage IoT entities remotely.

It might be important to calculate the cost for manual on-site management of IoT components in the IoT system. This cost can be compared to the possibility and cost to manage remotely and the needs for scalability of the IoT system. An IoT system lacking the possibility of remote management might be unmanageable even with a rather low number of IoT devices, if these devices are widespread geographically.

C.1.10 Network management and operation

IoT systems require network management. The form and purpose of network management and operation depend on network type, network ownership, and type of communication taking place over the network.

Some networks are managed as part of the IoT system – particularly the proximity networks connecting the IoT devices. Other networks, particularly the wide area networks, do not need to be managed as part of the IoT system. Such general-purpose networks are often operated by other organizations (like mobile phone networks). IoT network management spans both kinds of networks and assembles them into a coherent system that can serve the purposes of the IoT system. Where IoT systems use third-party general-purpose communication networks, their management and operational interfaces can be used, where available.

As for network capabilities the network management and operation capabilities influence the overall capabilities of the IoT system. IoT depending on a network that is established by a third party also places demand on that network for the overall trustworthiness of the IoT system. It is also important that the volume of data to be transferred by the network, load monitoring, and routing optimization capabilities are taken into account.

C.1.11 Real-time capability

Real-time capability is how an IoT system can handle real-time information with acceptable performance and how processing times and delays can be measured, recorded, and visualized. IoT systems often operate in real time; data about events in progress flow in continually and there can be a need to produce timely responses to that stream of events. This can involve stream processing: acting on the event data as they arrive, comparing against previous events and also against static data in order to react in the most appropriate way.

Larger volumes of data can cause delays not present at low data volumes. Also, the choice of network technology influences the real-time capability characteristics. Some network protocols, like 5G, are built for low latency, large data volumes, and better real-time performance. For IoT systems with high demands on the real-time characteristics, edge computing solutions are often required; that is, placing process power and process capabilities close to the IoT sensors.

C.1.12 Self-description

Self-description is when components of an IoT system list their capabilities in order to inform other IoT components or other IoT systems for the purposes of composition, interoperability, and dynamic discovery. Self-description includes interface specification, the capabilities of the IoT component, what types of devices can be connected to an IoT system, what kinds of service are made available by the IoT system, and the current state of the IoT system.

Self-description is useful when an IoT system interconnects with other IoT systems or those use cases where an IoT system benefits from being extended by the addition of new IoT devices. Self-description is also necessary for mobile devices and for devices that hibernate – both of which join and leave networks on a regular basis.

It is recommended to decide at an early stage what level of self-description capabilities is needed for the wanted business case, balancing costs compared with flexibility. The level of self-description also has implications on the choice of things like communication protocols and network technology.

C.1.13 Service subscription

Often IoT users subscribe to IoT services made available by IoT service providers. Then it is important that the IoT service provider establishes clear mechanisms for establishing and maintaining the subscriptions.

The IoT service providers make available a subscription process by which the IoT users can subscribe to a particular IoT service. The subscription process can include payments, plus a clear statement of any pre-requisites that apply to the IoT user. It can be the case that the IoT service involves the installation of IoT devices and the installation and configuration of software components. These devices and components are then typically provided or specified by the IoT service provider. Subscribing to a service and building a new IoT application can result in new safety requirements to the system. As the manufacturer of the IoT system during provisioning cannot foresee this use case, the responsibility of the safety requirement fulfilment lies with the subscriber. In some alternative cases, the IoT user can establish their own IoT service. In this case the IoT user has the burden of acquiring the necessary equipment and software. The user also has the subsequent responsibilities for operating and maintaining the IoT service. The possibility to offer service subscription might form the base for new business models by using service subscriptions instead of ordinary selling methods.

When implementing service subscriptions, it is often important to have in mind the life cycle of the services offered. When a service is upgraded or changed, it is often a good practice to keep the old service running for a certain time in parallel with the new service, to give the subscribers time to adopt to the new service offered.

C.2 Characteristics related to trustworthiness

C.2.1 Data characteristics – volume, velocity, veracity, variability, and variety

Data volume in an IoT system is often large, delivered at speed across network links, whose veracity it is important to validate (due to malfunctioning sensors), which can vary over time and can contain a wide variety of different data types from different IoT components.

This characteristic is also closely linked to the scalability capability of the IoT system. It is crucial to analyse all bottle necks to ensure the data characteristics of all parts in the IoT system are appropriate for the business needs. These issues might be more complex and difficult to spot and calculate in the complex environment of many IoT systems.

C.2.2 Protection of personally identifiable information (PII)

As soon as an IoT system measures information directly or indirectly related to physical persons it deals with sensitive information. Sensitivity extends to all PII from which sensitive PII can be derived, whether through aggregation, analysis, or other means. Protection of PII is a legal or regulatory requirement in most jurisdictions. Protection of PII is a general requirement and is governed by a series of principles which are described in ISO/IEC 29100 [14]. It is important to apply these principles in any IoT system that is processing PII. System owners in many jurisdictions are required to disclose a data breach. In the event of a compromise, it is important to ensure that they can identify the data that was compromised and report to local agencies. When there are requirements to disclose and report a data breach, it is particularly important to implement the possibility to discover such data breaches at all levels and stages of an IoT system processing PII.

This document uses PII as defined by ISO/IEC 27018:2019, 3.2: "any information that (a) can be used to establish a link between the information and the natural person to whom such information relates, or (b) is or can be directly or indirectly linked to a natural person" [15].

Breaches not only damage the business brand but can also result in severe costly penalties. IoT data is often transferred and combined with other datasets. Therefore, it is crucial to limit, or at least have very good control over, how PII are forwarded. It is also crucial to carefully consider how the PII data can cause harm or break regulatory rules if combined with other datasets.

C.2.3 Flexibility

Flexibility is the capability of an IoT system, service and device or other component to provide a varied range of functionality, depending on need or context.

History and experience tell that while there are exceptions, the economic and functional sweet spot for flexibility is somewhere in the middle. At one end of the spectrum are the extremes of a dedicated single-purpose component. At the other end of the spectrum is a massively capable, programmable, extensible, "all things to all people" general-purpose component.

Bibliography

[1] ISO/IEC/IEEE 42010:2022, *Software, systems and enterprise – Architecture description*

[2] ISO/IEC JTC1 draft standing document. Best practices and guidelines for reference architecture standards, March 2023

[3] ISO/IEC 20924, *Information technology – Internet of Things (IoT) – Vocabulary*

[4] IEC 62443 (all parts), *Security for industrial automation and control systems*

[5] ISO/IEC 27000, *Information technology – Security techniques – Information security management systems – Overview and vocabulary*

[6] ISO/IEC TS 30149, *Internet of Things (IoT) – Trustworthiness principles*[2]

[7] IEC PAS 63088, *Smart manufacturing – Reference architecture model industry 4.0 (RAMI4.0)*

[8] CEN-CENELEC-ETSI Smart Grid Coordination Group. Smart Grid Reference Architecture. https://www.cencenelec.eu/media/CEN-CENELEC/AreasOfWork/CEN-CENELEC_Topics/Smart%20Grids%20and%20Meters/Smart%20Grids/reference_architecture_smartgrids.pdf

[9] Industry IoT consortium pattern library, https://www.iiconsortium.org/patterns/

[10] IEC 61360, *IEC Common Data Dictionary (CDD)*, available at http://cdd.iec.ch/

[11] ISO 13584-42:2010, *Industrial automation systems and integration – Parts library – Part 42: Description methodology: Methodology for structuring parts families*

[12] ECLASS, https://eclass.eu/en

[13] ISO/IEC 15045-1:2004, *Information technology – Home Electronic System (HES) gateway – Part 1: A residential gateway model for HES*

[14] ISO/IEC 29100, *Information technology – Security techniques – Privacy framework*

[15] ISO/IEC 27018:2019, *Information technology – Security techniques – Code of practice for protection of personally identifiable information (PII) in public clouds acting as PII processors*

[2] Under preparation. Stage at the time of publication: ISO/IEC DTS 30149:2023.

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

3, rue de Varembé
PO Box 131
CH-1211 Geneva 20
Switzerland

Tel: + 41 22 919 02 11
info@iec.ch
www.iec.ch