

# Conceptualization of IoT architectures

Gaetanino Paolone<sup>1</sup>, Romolo Paesani<sup>2</sup>, Jacopo Camplone<sup>1</sup>, Andrea Piazza<sup>1</sup>, Paolino Di Felice<sup>3</sup>

<sup>1</sup>B2B S.r.l., Teramo, Italy

<sup>2</sup>Gruppo SI S.c.a.r.l., Teramo, Italy

<sup>3</sup>Department of Industrial and Information Engineering and Economics, University of L'Aquila, L'Aquila, Italy

## Article Info

### Article history:

Received Jun 17, 2024

Revised Oct 14, 2024

Accepted Nov 19, 2024

### Keywords:

Architecture viewpoint

Architecture description

Framework

Internet of things

Stakeholder perspective

Stakeholder

Standard

## ABSTRACT

Although there is a large interest about internet of things (IoT) architectures, still there is no consensus on their conceptualization in the extant literature. This lack of information in conceptualization is problematic because it hampers the deep understanding of the appeared proposals, as well as the adoption of a shared workflow by the involved architects of these systems. Thus, a concise and agreed-upon conceptualization of IoT architectures is called for. This paper aims at giving a contribution on the topic. We start by reviewing the available standards, then, in light of their suggestions, a workflow to be followed in the definition of the architecture descriptions (ADs) of IoT systems is detailed and, in addition, a sample case study, which implements that workflow, is proposed. The contributions are sufficiently abstract to be applicable also to the description of the architecture of artificial intelligence of things (AIoT) systems.

*This is an open access article under the [CC BY-SA](#) license.*



## Corresponding Author:

Paolino Di Felice

Department of Industrial and Information Engineering and Economics, University of L'Aquila

L'Aquila, Italy

Email: [paolino.difelice@univaq.it](mailto:paolino.difelice@univaq.it)

## 1. INTRODUCTION

This section starts by introducing the internet of things (IoT). Then, the role of architecture description (ADs) is emphasized as the means to manage the complexity of IoT-based systems. Gap identification and contributions of the study are successively given; while paper's structure ends the section.

The IoT is a network of physical devices, interfaces, and other items embedded with sensors, actuators, electronics, and connectivity. The number of IoT connected devices worldwide in 2023 was 15.14 billions (source: <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/> -accessed 11 April, 2024). An IoT infrastructure includes the following basic components:

- Sensors (they gather real-time data from the environment and convert it into a digital signal),
- Microcontrollers (they process and manage the data collected by the sensors),
- Communication modules (they transmit the data over the network), and
- Cloud (it offers infrastructures, servers and storage, needed for the data processing).

Data is only useful if it creates an action. To make data truly actionable, it needs to be supplemented with context. Artificial intelligence (AI) and IoT together (shortly, artificial intelligence of things (AIoT)) are the context [1], [2]. Combining IoT with AI technologies can create "smart machines" able to make decisions with little or no human intervention. The benefits deriving from the marriage of AI and IoT have been highlighted for all IoT application domains. Hereafter we will talk generically of IoT, but with few exceptions, that will be clear from the context, the reasonings embrace also the AIoT.

The complexity of IoT systems has grown to an unprecedented level. This has created new opportunities, but at the prize of a long list of challenges for the stakeholders that create and/or use these systems. Stakeholders interests about these systems are expressed as concerns about them. Concerns are the result of the stakeholders' perspectives, the latter originating from domain knowledge, skill, responsibility and role played in the organization.

In academia and industry, architecting IoT systems is usually proposed to help manage the complexity faced by the involved stakeholders. This claim is proved by the high number of distinct IoT architectures that have been already published. For instance, Alshohoumi *et al.* [3] analyzed 148 studies and identified sixteen different architectures. Because of the high number of proposals, IoT architectures have been classified according to: (i) the provenance (academia or industry); (ii) the application domain; and (iii) the style (layered, service-oriented, middleware-oriented, and computing-paradigm-oriented) [4]-[6]. As a quite obvious consequence of the variety of available distinct solutions, the terms adopted in the description of IoT architectures vary considerably from one technological solution to the other [7].

This issue is confirmed by the work by Wang *et al.* [8], carried out a deep review of 20 highly cited papers on the basic ("functional") components of the IoT. 71 distinct sentences resulted to be used to denote those components, but there were many overlapping and duplicate words among them. The 71 sentences were divided into the following five independent sets: smart device, perception, cloud, transportation, and application. Accordingly, the authors concluded that the predominant IoT systems are composed of five parts: a device layer, a perception layer, a cloud layer, a transport layer, and an application layer.

The examination of the extant literature on architectures highlights two opposing attitudes. In one (expressed by the minority of scholars), it is dreamed a unique IoT architecture which is acceptable for all applications [9], [10]. In the other, the claim is that a single architecture is not enough for abstracting the diverse needs of all the potential IoT applications [4]-[6], [11]-[16]. As previously said, most published research studies belonging to the IoT domain talk about architectures of these heterogeneous systems, but too often there is no information on the stakeholders the proposal is intended to reach, on the concerns the proposal addresses, and neither, and even worse, on the process how the proposal is built. In this paper, we use the term conceptualization to refer to such a process. Collins English Dictionary defines conceptualization as "The process of forming a concept out of observations, experience, and data." The position expressed in this work is in line with the prevailing one mentioned above, but the basic assumption of our study is that the AD must be guided by a deterministic workflow that implements the stages envisaged by an architectural framework shared by the scientific community. Moreover, the steps of the workflow must highlight the stakeholders to whom the description of the architecture is addressed and the concerns that the latter intercepts. Specifically:

- We recall the fundamentals of the ISO/IEC/IEEE 42010 standard published in 2022 [17] that has replaced the ISO/IEC/IEEE 42010:2011;
- Then, three distinct proposals based on [17] are summarized: the ISO/IEC 30141:2018 standard elaborated by the international organization for standardization (ISO) and the international electrotechnical commission (IEC) joint technical committee 1: information technology [18]; the IEEE Std 2413-2019 standard elaborated by the IEEE SA standards board [19], and the report by the industrial internet consortium (IIC) [20].
- The comparative study of the four documents allowed us to highlight how the recommendations in the ISO/IEC/IEEE 42010 standard have been implemented in the subsequent three proposals.
- It was, also, possible to define the deterministic workflow comprising the stages envisaged by the architectural framework shared among the recalled four proposals.
- Finally, a case study, broken down into the steps that make up the defined workflow, is sketched.

The paper is structured as follows. Section 2 recalls the related work; while section 3 presents the research method of this study; then section 4 summarizes the results. The enucleation of the workflow that lists the sequence of stages in the production of the artifacts that describe the IoT system to be developed is part of the section. Section 5 applies the workflow to a case study; eventually, section 6 ends the paper.

## 2. RELATED WORKS

This section recalls three studies that before ours have pointed out the need to bring order to the dispersed body of knowledge about IoT architectures. We didn't spend further time to search for other studies

that could have expressed the same position, simply because we considered the relevance of the three selected papers sufficient to motivate our work. The common basic assumption of the three selected works is that to understand the available proposals and maps similarities and differences among them, it is necessary to adopt an IoT standard reference architecture. The need to refer to a shared workflow in the construction of architecture views that intersect specific stakeholders' concerns is not mentioned in those studies. In other words, they have taken a direction independent of the philosophy embedded in the ISO/IEC/IEEE 42010 standard [17], on which our study is based.

Muccini *et al.* [4] carried out a systematic mapping study on IoT architectural styles in order to identify characteristics and publication trends. They selected 63 papers out of about 2,300 possible choices. Then, they classified the architectural styles as a set of abstract IoT reference architectures.

Di Martino *et al.* [7], provide a review of the most common IoT architectural solutions available up to 2018. Extant commercial proposals and two standards were taken into account in the study. In detail, authors adopted the layered-functional architecture proposed in [21] as a reference architecture to analyze and compare the reference architectures introduced, respectively, in the standards [18], [22].

Ameyed *et al.* [23], authors carried out a qualitative and quantitative comparative analysis of four commercial architectures (namely, Intel, Microsoft, Cisco, and Google) against the Domain-based model part of the ISO/IEC 30141 reference IoT architecture [18]. The assessment was carried out by means of an evaluation framework based on quantitative metrics and scoring methods proposed by authors. The final aim of the study was to map the similarities and differences of the commercial solutions.

### 3. RESEARCH METHOD

The method utilized in this study involved a detailed examination of previous efforts to define guidelines for promoting the conceptualization of the IoT architecture. The workflow of the research method looking for issued standards' proposals is illustrated in Figure 1. This section comprises four sub-sections as many are the extant proposals, [17]-[20]. Each sub-section provides, in sequence, a primer of the conceptual framework given in those documents. The comparison of the four proposals is the subject of next section.

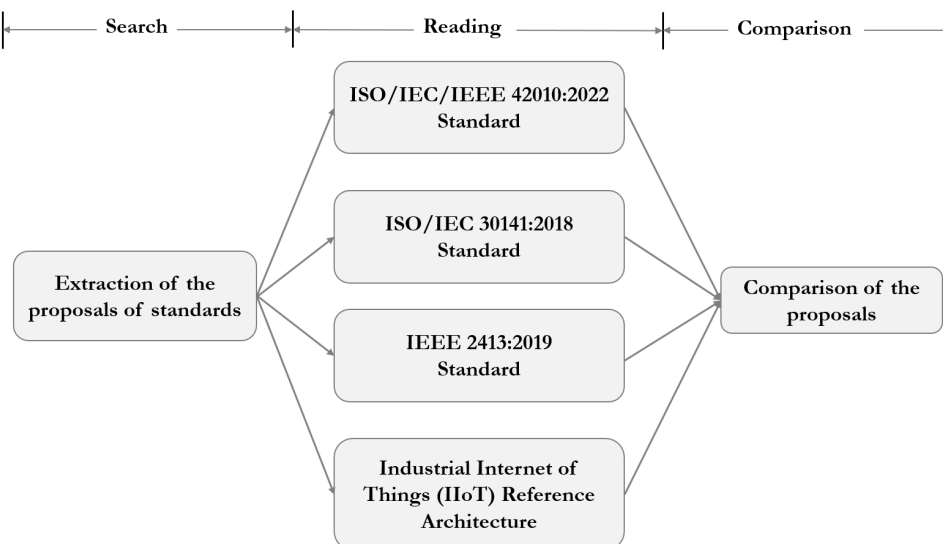


Figure 1. The research method workflow

#### 3.1. The ISO/IEC/IEEE 42010:2022 standard

This standard introduces the guidelines useful for describing architectures of complex systems (the latter generically called entity of interest in the document). Stakeholders, stakeholder perspectives, stakeholder concerns, ADs are the main concepts at the bottom of such a standard. Hereafter, they are briefly recalled.

### 3.1.1. Architecture descriptions

The architecture of an entity of interest, expressed by one or more ADs, assists in understanding the basic concepts or properties of the entity, referring, for instance, to its structure and behavior. ADs are useful to improve communication and cooperation among stakeholders. Figure 2 depicts the relationships among the basic concepts this standard relies on. The entry point in the conceptual graph is the stakeholder concept.

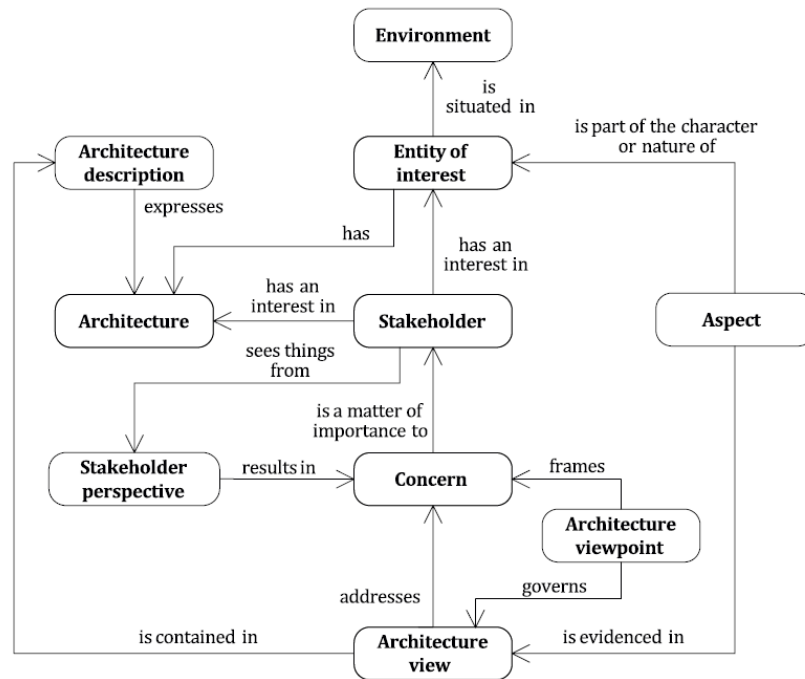


Figure 2. The graph about the main concepts (the nodes) and their relationships (the edges) in the ISO/IEC/IEEE 42010:2022 standard [17]

A stakeholder is an individual or an organization having an interest or a right in an entity of interest. End users, operators, owners, suppliers, architects, developers, builders, maintainers, certifying agencies are examples of stakeholders. Within this paper, IoT systems are the entity of interest. Entity of interests are situated in an environment; the latter can have various influences upon them [17]. Interest in an entity comprises interest in its environment, requirements, architecture, design, implementation, operation, and life cycle.

Aspects, concerns and stakeholder perspectives allow to express such interests. A stakeholder perspective is a way of thinking about an entity of interest. Relevant perspectives include: strategic, organizational, operational, logical, physical and technological ones. Perspectives result in concerns. A concern is a matter of relevance to a stakeholder. Architecture viewpoints comprise conventions necessary for the creation, interpretation and use of architecture views in order to frame concerns. An architecture view constitutes a portion of an AD, the latter being an artifact that expresses an architecture to be provided to the stakeholders. An AD may contain more architecture views. Organizing ADs into architecture views governed by architecture viewpoints allows the separation of concerns based on stakeholders perspectives, providing, at the same time, an integrated view of the whole entity.

### 3.1.2. Architecture description frameworks

An ADF regards a set of best practices for creating, interpreting, analyzing and using ADs within a particular domain of interest. In other words, according to the ISO/IEC/IEEE 42010:2022 standard, an ADF is the umbrella under which ADs must be done. Figure 3 shows the UML class diagram collecting the concepts that all together are referred to as ADF in [17], namely: domain of interest, concerns, stakeholders, viewpoints, and model kinds. Model kind is a set of conventions for the formalization of concerns, while a Model is the artifact produced by applying a specific model kind (e.g., UML). An architecture viewpoint may specify several model kinds.

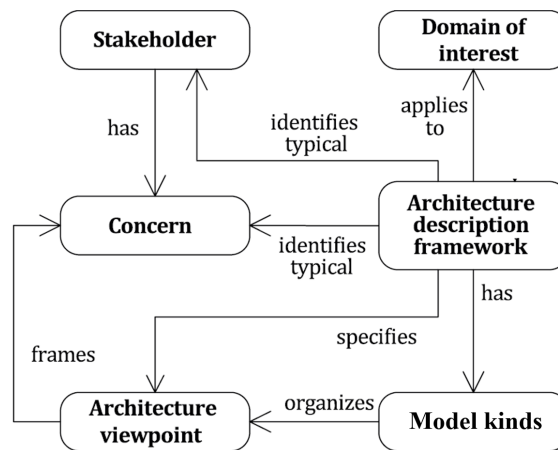


Figure 3. The conceptual model of the ADF defined in [17]

Figure 4 shows a meta example about the basic concepts part of the ISO/IEC/IEEE 42010 standard. Two stakeholders are interested in the same entity of interest (a smart city). Both have perspective viewpoints about the entity of interest. From each perspective originates an architecture viewpoint that, in turn, gives rise to an architecture view.

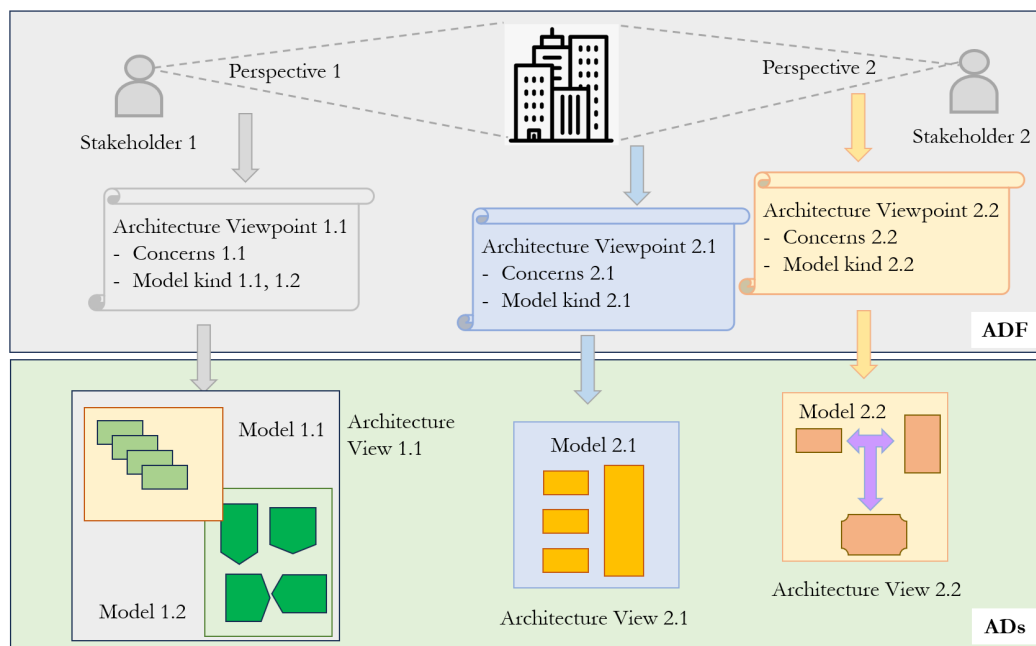


Figure 4. A meta example depicting the mapping from the ADF to ADs

### 3.2. The ISO/IEC 30141:2018 standard

This standard has the following merits: (i) it is technology-neutral; (ii) it gives a clear picture of IoT systems to the involved stakeholders; (iii) it simplifies the communication between them. Overall, ISO/IEC 30141:2018 [18] conveys useful advices to the IoT architect to build his own ADs as meant in the previous subsection and then actual systems. IoT characteristics, conceptual model, and reference model are the constituent pillars of this standard (Figure 5). They are recalled below.

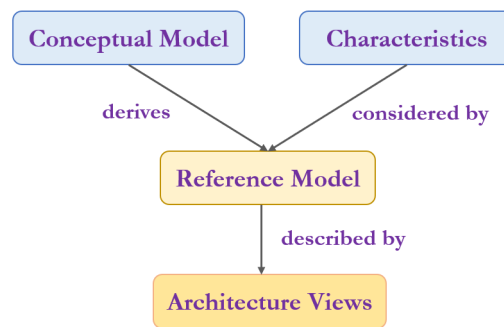


Figure 5. The structure of the ISO/IEC 30141:2018 standard

Table 1 collects the most relevant IoT characteristics. The conceptual model abstracts these characteristics. It is presented by means of UML class diagrams concerning concepts as: virtual/physical entities; IoT devices; IoT users; IoT gateways; the network, the services. The reference model is presented from two complementary perspectives: the first (perspective) is entity-based (IoT users, IoT gateways, IoT devices, networks, physical entities are examples of entities), while the second one is domain-based. The identified domains are: user domain, operations/management domain, application/service domain, resource access and interchange domain, sensing and controlling domain, and physical entity domain.

Table 1. Characteristics of IoT systems according to [18]

Trustworthiness	Architecture	Functional
Availability	Composability	Accuracy
Confidentiality	Functional and management capability separation	Auto-configuration
Integrity	Heterogeneity	Compliance
Protection of personally identifiable information	Highly distributed systems	Content-awareness
Reliability	Legacy support	Context-awareness
Resilience	Modularity	Data characteristics: volume, velocity, ..
Safety	Network connectivity	Discoverability
	Scalability	Flexibility
	Shareability	Manageability
	Unique identification	Network communication
	Well-defined components	Network management and operation
		Real-time capability
		Self-description
		Service subscription

From this three pillars, four architectural views are discussed (Figure 6): a functional view; a system deployment view; a networking view; and a usage view. Briefly,

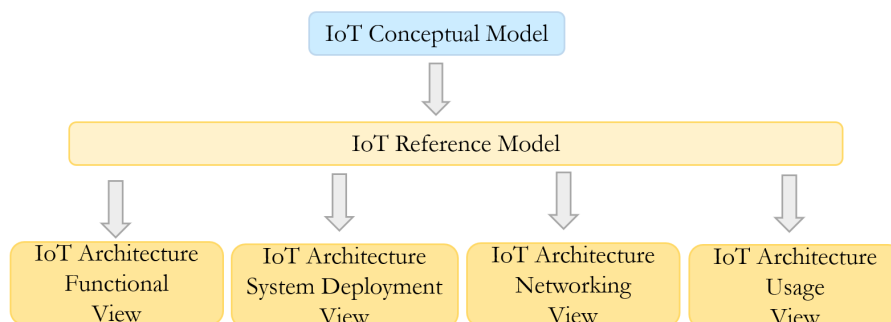


Figure 6. The architectural views discussed in [18]

- The functional view is a technology-agnostic view of the components necessary to form an IoT system.
- The system deployment view describes the actual components that form an IoT system, namely devices, subsystems, and networks.
- The networking view describes the principal communications networks which are involved in IoT systems and the entities with which they connect.
- The usage view focuses on how the IoT system is developed, tested, operated, and used from a user perspective.

The expression “IoT reference architecture”, used in [18], corresponds to the ADF in [17]. In fact, it encapsulates the whole architectural framework described in the standard as a generic conceptual template from which a certain number of context-specific IoT architecture views, governed by architecture viewpoints, can be defined. We do not use this expression in this work to prevent misunderstandings, since the expression “Reference Architecture” seems to convey the message that exists just one IoT architecture, at the opposite it has been remarked (subsection 3.1.) that there is at least one architecture view for each Architecture viewpoint.

### 3.3. The IEEE 2413:2019 standard

Such a standard [19] introduces an architecture framework for the IoT which conforms to the international standard ISO/IEC/IEEE 42010:2011 and hence to the ISO/IEC/IEEE 42010:2022. The architecture framework is motivated by concerns commonly shared by IoT system stakeholders across the following six relevant domains: smart manufacturing, smart grid, smart buildings, smart cities, smart healthcare, intelligent transport systems. The concerns are elaborated as a set of architecture viewpoints that form the body of the framework description. A peculiarity feature of standard [19] is the emphasis it puts on pointing out stakeholders viewpoints and concerns. 15 stakeholders (Table 2), 13 viewpoints (Table 3), and 62 concerns (see Table 2; p.37 in [19]) are listed. From each viewpoint, it is possible to build an architecture view, in line with [17]. IEEE [19] describes in detail the 13 viewpoints (pp. 44–163). Hereafter, we restrict our attention to the conceptual viewpoint.

Table 2. The stakeholders

Stakeholder	Stakeholder (1)	Stakeholder (2)
Acquirers	Maintainers	Suppliers
Assessors	Operators	Support staff
Builders	Owners	System administrators
Communicators	Production engineers	Testers
Developers	Regulators	Users

Table 3. The viewpoints

Viewpoint	Viewpoint (1)	Viewpoint (2)
Conceptual viewpoint	Function viewpoint	Privacy and trust viewpoint
Compatibility viewpoint	Threat model viewpoint	Collaboration viewpoint
Lifecycle viewpoint	Security and safety monitoring viewpoint	Computing resources viewpoint
Communication viewpoint	Access control viewpoint	
Information viewpoint	Adequate design for required security viewpoint	

Conceptual viewpoint concerns the definition of a vocabulary and semantics about IoT systems. In the process of building architecture views, it is necessary to build such a view in order to ensure that the involved stakeholders employ a shared language when talking about these heterogeneous systems. The conceptual viewpoint comprises six complementary models: the entity model, the system model, the intent model, the component model, the component capability model, and the representation model. It is worth noting that the way to construct the conceptual viewpoint is not unique. We will come back to this aspect in next section.

### 3.4. The industrial internet of things reference architecture

This report details an industrial internet architecture framework (IIAF) based on the ISO/IEC/IEEE 42010:2011 standard. Then, an industrial internet reference architecture (IIRA) is built according to the IIAF. IIRA abstracts common characteristics, features and patterns from several industrial domains. Four viewpoints

are discussed in [20]: business viewpoint, usage viewpoint, functional viewpoint, and implementation viewpoint. In the intention of the industrial internet consortium, system architects may adopt these viewpoints as starting point and then extend them by defining additional viewpoints to organize system concerns based on the specific system requirements. For example, in ref. [24] authors specialize the three-tier architecture pattern that is part of the implementation viewpoint in [20].

#### 4. RESULTS AND DISCUSSION

This study reviewed four well-known documents about the conceptualization of IoT architectures [17]-[20]. Each document underlined the relevance of the topic given the increasing role that the IoT technology plays in the solution of a huge number of every-day problems. Unfortunately, there is still no consensus on the conceptualization workflow. Moreover, despite there are earlier studies referring to some of the four proposals (see section 2), so far, it has not explicitly investigated the correspondence of their basic notions. The present study fills both these gaps, as it is detailed in the following two sub-sections.

##### 4.1. Concepts correspondence

Below, it is described the correspondence of the main concepts of the ADFs in [17] into the three alternatives architecture frameworks recalled in sub-sections 3.2, 3.3, and 3.4. In detail, Tables 4-6 compare, in sequence, the ISO/IEC/IEEE 42010:2022 [17] against ISO/IEC 30141:2018 [18], IEEE 2413:2019 [19], and the IIRA [20].

Table 4. ISO/IEC/IEEE 42010:2022 vs. ISO/IEC 30141:2018

ISO/IEC/IEEE 42010:2022	ISO/IEC 30141:2018
Entity of interest	IoT systems
Stakeholders	Users, developers, architects, ...
Stakeholder perspectives/concerns	Get an overview of basic entities (of IoT systems) Get an overview of tasks to be performed
Architecture viewpoint	Entity-based viewpoint Domain-based viewpoint
Architecture view	IoT RA functional view IoT RA system deployment view IoT RA communications view IoT RA usage view

Table 5. ISO/IEC/IEEE 42010:2022 vs. IEEE 2413:2019

ISO/IEC/IEEE 42010:2022	IEEE 2413:2019
Entity of interest	IoT systems
Stakeholders	15 stakeholders are listed
Stakeholder perspectives/concerns	62 concerns are listed
Architecture viewpoint	13 viewpoints are listed
Architecture view	Architecture view

Table 6. ISO/IEC/IEEE 42010:2022 vs. [20]

ISO/IEC/IEEE 42010:2022 [17]	RA stands for reference architecture [20]
Entity of interest	End users, developers, architects, ...
Stakeholder perspectives/concerns	Get an overview of basic entities (of IoT systems) Get an overview of tasks to be performed
Architecture viewpoint	Business viewpoint Usage viewpoint Functional viewpoint Implementation viewpoint
Architecture view	IIoT RA business view IIoT RA Usage view IIoT RA IIoT RA functional view IIoT RA implementation view

It is worth noting that the way to construct the conceptual viewpoint is not unique. For example, Table 7 shows that in the standard [19], such a viewpoint employs six UML models, while in the standard [18], the conceptual model is composed of a number of key properties that an IoT system typically exhibits (briefly called characteristics - Table 1) and a certain number of UML class diagrams describing the key concepts characterizing these systems.

Table 7. Instantiation of the Conceptual viewpoint in [18], [19]

The standard	Implementation of the conceptual viewpoint	Modeling language
ISO/IEC 30141:2018 [18]	IoT characteristics (Table 1)	
	A certain number of class diagrams	UML
IEEE Std 2413-2019 [19]	Entity model	UML
	System model	UML
	Intent model	UML
	Component model	UML
	Component capability model	UML
	Representation model	UML

#### 4.2. Workflow for architecture description

Table 8 lists the stages of the workflow for AD that results from the ISO/IEC/IEEE 42010:2022 standard. Each stage has a name (second column) and a justification. The underlying hypothesis is that in our case the entity of interest concerns the IoT, while the domain remains generic.

Table 8. Stages of the recommended workflow (AD stands for architecture description)

Step	Name	Comment
1	Explanation of the AD purpose	An AD must include a statement of its intended purpose.
2	Identification of stakeholders	An AD must identify the stakeholders having concerns about the architecture of the IoT system and consistent with the purpose of the AD.
3	Identification of stakeholder perspectives	An AD must identify stakeholder perspectives about the architecture of the IoT system and consistent with the purpose of the AD.
4	Identification of concerns	For each identified perspective, an AD must enumerate the pertinent concerns from among the list of concerns. An AD must associate each identified concern with the stakeholders holding that concern.
5	Enucleation of architecture viewpoints	For each identified perspective must be established the architecture viewpoints necessary to frame the identified concerns.
6	Production of the architecture views	To each identified architecture viewpoint, at least an architecture view that addresses the concern has to be bounded.

### 5. CASE STUDY

This section elaborates a sample case study that implements the stages listed in the previous section. The IoT is the entity of interest, while the application domain is kept undefined on purpose. Within the case study, the intended purpose of the AD is limited to produce the UML class diagram about the component capability model of an IoT system that adopts edge computing. The involved stakeholders are listed in Table 9.

Table 9. Stakeholders and their role

Stakeholder	Role/Responsibility
Users	Collaborate in the definition of the IoT system requirements. Use it once deployed.
Engineers	Design the hardware and software necessary to run the IoT system according to the requirements.
Developers	Deploy the IoT system.
Operators and maintainers	Run the IoT system once it has been deployed. Manage the evolution.
Owners	Derive the benefits of the solution when in use.

#### 5.1. The concerns

We list relevant requirements to be taken into consideration when designing and deploying an IoT system:

- Data volume versus bandwidth: IoT sensors can produce more data than is economically feasible to transmit to the cloud because of communication costs that represent a relevant quota of the expenses. So, the issue is to find a sustainable tradeoff.
- Performance constraints: network latency is an obstacle to meet the need of real-time reaction posed by many actual applications. processing the sensed data as close as possible to the sensors has been proved to be the solution.
- Anticipate decision-making: it becomes possible by introducing intelligence at the source nodes of the IoT system.
- Privacy constraints: to limit the risk that the data crossing the network moving towards the cloud would be stolen, it is necessary to process it near to the source.
- Intermittent connectivity: when devices and sensors are in locations with only intermittent connectivity, they need local data processing and decision-making in order to keep operating. Reducing the time when the IoT device is connected to the network has the extra benefit of preserving the battery life of sensors.

It has been pointed out that an edge-computing-oriented solution (Figure 7) may address all the previous five concerns. Edge computing is performed on a platform at the network edge near the things, integrating network, computing, storage, and application main capabilities adding edge intelligent services [1], [25]-[27]. In particular, using caching and/or local algorithms to pre-process the data at the edge reduces the communication cost. This feature supports autonomous operation as well, especially when the connectivity is intermittent.

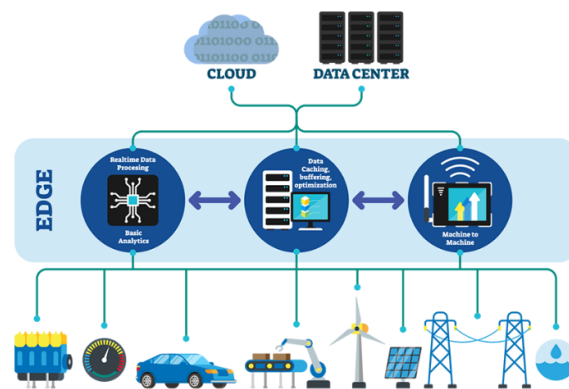


Figure 7. Edge computing scenario

As it emerges from the ISO/IEC/IEEE 42010 standard, and explained in detail by the standard [19], to achieve an effective description of the architecture of the entity of interest it may be useful to adopt multiple viewpoints. For the purposes of this example, we limit the attention to the conceptual viewpoint (Table 3). Among the 6 models that it provides [19], below, we focus on the component capability model (Table 7).

## 5.2. Conceptual model: IoT component capability model

The UML class diagram of Figure 8 shows the pertinent capability classes of an IoT system. A brief description of each of them follows. Preliminary, the diagram shows that an IoT system may be composed of an arbitrary number of IoT components. The following capabilities apply to each of them.

- Sensing capability: provides a value of a property of the physical entity of interest in the form of digital data. Data gained by sensors may be provided to other IoT components through the component's network interface for processing and storage. Examples of observations concern temperature, position, and audio.
- Actuating capability: provides the ability to make a change in the physical world, based on a digital input to the component. An electronic door lock is an example of the actuating capability.
- Data storing capability: provides the ability to store and retrieve data. Databases and data brokers (such as the message queue telemetry transport broker) are examples of data-storing capabilities.
- Data-transferring capability: provides the ability to transmit data from one location to another. Data networks based on ethernet and long-term evolution (LTE) are examples of data-transferring capabilities.

- Data-processing capability: provides the ability to transform data based on an algorithm. Data aggregation, data analytics, and predictive analysis are examples of processing.
- Supporting capability: concerns additional functionality supporting the functioning of the IoT component or IoT system. Time synchronization, data encryption, authentication, orchestration, and remote component management are examples of supporting capabilities.
- Network interface capability: concerns the ability to interface with a communication network. Every IoT component shall have at least one network interface capability. Ethernet adapter interface capability and LTE radio interface capability are specific examples.
- Human UI capability: concerns the ability for the component to exchange information directly with people. Displays, touch screens, audio speakers, and microphones are examples of human UI capabilities.
- Application interface capability: provides the ability for an IoT component to communicate with another IoT component through an application. Application programming interface (API) is a widely-used type of application interface.

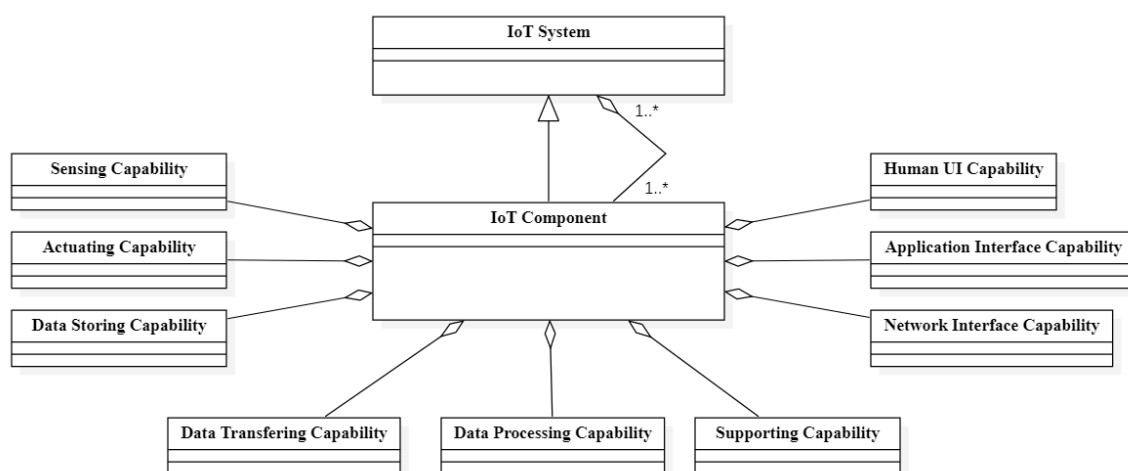


Figure 8. Class diagram of the capabilities of an IoT system adopting edge computing

It remains to be noted that to bring to light the many aspects relevant for IoT systems, it is often necessary to also detail other architecture viewpoints among those listed in Table 3. Making the appropriate choices is a responsibility of IoT architects. Standard [19] proposes interesting examples to bring hints from.

## 6. CONCLUSION

This paper elaborated the notion of conceptualization of IoT architectures meant as the workflow to be followed in the construction of IoT ADs that highlight the stakeholders to whom the architecture is addressed and the concerns that the latter intercepts. The detailed workflow comes from the recommendations of the ISO/IEC/IEEE 42010:2022. The adoption of a discipline in the construction of IoT ADs brings the positive effect of facilitating the cooperation among the stakeholders taking part in the requirement definitions, system design and deployment, their operation and evolution over time, and their documentation.




## REFERENCES

- [1] O. Debauche, S. Mahmoudi, S. A. Mahmoudi, P. Manneback, and F. Lebeau, "A new edge architecture for AI-IoT services deployment," *Procedia Computer Science*, vol. 175, pp. 10–19, 2020, doi: 10.1016/j.procs.2020.07.006.
- [2] T. W. Sung, P. W. Tsai, T. Gaber, and C. Y. Lee, "Artificial intelligence of things (AIoT) technologies and applications," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, Jan. 2021, doi: 10.1155/2021/9781271.
- [3] F. Alshohoumi, M. Sarrah, A. Al-Hamadani, and D. Al-Abri, "Systematic review of existing IoT architectures security and privacy issues and concerns," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 7, pp. 232–251, 2019, doi: 10.14569/IJACSA.2019.0100733.




- [4] H. Muccini and M. T. Moghaddam, "IoT architectural styles: a systematic mapping study," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11048 LNCS, pp. 68–85, 2018, doi: 10.1007/978-3-030-00761-4\_5.
- [5] S. P. Singh, V. Kumar, A. K. Singh, and S. Singh, "A survey on internet of things (IoT): layer specific vs. domain specific architecture," in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 44, 2020, pp. 333–341.
- [6] S. Bansal and D. Kumar, "IoT ecosystem: a survey on devices, gateways, operating systems, middleware and communication," *International Journal of Wireless Information Networks*, vol. 27, no. 3, pp. 340–364, Sep. 2020, doi: 10.1007/s10776-020-00483-7.
- [7] B. Di Martino, M. Rak, M. Ficco, A. Esposito, S. A. Maisto, and S. Nacchia, "Internet of things reference architectures, security and interoperability: a survey," *Internet of Things*, vol. 1–2, pp. 99–112, Sep. 2018, doi: 10.1016/j.iot.2018.08.008.
- [8] J. Wang, M. K. Lim, C. Wang, and M.-L. Tseng, "The evolution of the internet of things (IoT) over the past 20 years," *Computers & Industrial Engineering*, vol. 155, p. 107174, May 2021, doi: 10.1016/j.cie.2021.107174.
- [9] C. C. Sobin, "A survey on architecture, protocols and challenges in IoT," *Wireless Personal Communications*, vol. 112, no. 3, pp. 1383–1429, Jun. 2020, doi: 10.1007/s11277-020-07108-5.
- [10] A. A. Laghari, K. Wu, R. A. Laghari, M. Ali, and A. A. Khan, "Retracted article: a review and state of art of internet of things (IoT)," *Archives of Computational Methods in Engineering*, vol. 29, no. 3, pp. 1395–1413, May 2022, doi: 10.1007/s11831-021-09622-6.
- [11] P. Sethi and S. R. Sarangi, "Internet of things: architectures, protocols, and applications," *Journal of Electrical and Computer Engineering*, vol. 2017, pp. 1–25, 2017, doi: 10.1155/2017/9324035.
- [12] H. Kaur and R. Kumar, "A survey on internet of things (IoT): layer-specific, domain-specific and industry-defined architectures," in *Advances in Intelligent Systems and Computing*, vol. 1086, 2021, pp. 265–275.
- [13] W. Kassab and K. A. Darabkh, "A–Z survey of internet of things: architectures, protocols, applications, recent advances, future directions and recommendations," *Journal of Network and Computer Applications*, vol. 163, p. 102663, Aug. 2020, doi: 10.1016/j.jnca.2020.102663.
- [14] M. Lombardi, F. Pascale, and D. Santaniello, "Internet of things: a general overview between architectures, protocols and applications," *Information*, vol. 12, no. 2, p. 87, Feb. 2021, doi: 10.3390/info12020087.
- [15] B. Mazon-Olivo and A. Pan, "Internet of things: state-of-the-art, computing paradigms and reference architectures," *IEEE Latin America Transactions*, vol. 20, no. 1, pp. 49–63, Jan. 2022, doi: 10.1109/TLA.2022.9662173.
- [16] M. Adam, M. Hammoudeh, R. Alrawashdeh, and B. Alsulaimy, "A survey on security, privacy, trust, and architectural challenges in IoT systems," *IEEE Access*, vol. 12, pp. 57128–57149, 2024, doi: 10.1109/ACCESS.2024.3382709.
- [17] "ISO/IEC/IEEE 42010:2022 Software, systems & enterprise - Architecture description." <https://www.iso.org/standard/74393.html> (accessed May 05, 2024).
- [18] "ISO/IEC 30141:2018, internet of things (IoT)-reference architecture." <https://www.iso.org/obp/ui/#iso:std:iso-iec:30141:ed->
- [19] IEEE, "IEEE P2413 standard for an architectural framework for the internet of things (IoT)," *IEEE Standards Association*. IEEE, Piscataway, NJ, USA, May 21, 2017, doi: 10.1109/IEEESTD.2020.9032420.
- [20] M. Guizani, "The industrial internet of things," *IEEE Network*, vol. 33, no. 5, pp. 4–4, Sep. 2019, doi: 10.1109/MNET.2019.8863716.
- [21] E. Borgia, "The internet of things vision: key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1–31, 2014, doi: 10.1016/j.comcom.2014.09.008.
- [22] *Overview of the Internet of things. Recommendation ITU-T Y.2060* ITU-T Y-Series Recommendations, Global Information Infrastructure, Internet Protocol Aspects and Next Generation Networks, 15 June 2012 <https://www.itu.int/rec/T-REC-Y.2060-201206-I>
- [23] D. Ameyed, F. Jaafar, F. Petrillo, and M. Cheriet, "Quality and security frameworks for IoT-architecture models evaluation," *SN Computer Science*, vol. 4, no. 4, p. 394, May 2023, doi: 10.1007/s42979-023-01815-z.
- [24] A. P. D. de Araújo et al., "General system architecture and COTS prototyping of an AIoT-enabled sailboat for autonomous aquatic ecosystem monitoring," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 3801–3811, Feb. 2024, doi: 10.1109/JIOT.2023.3324525.
- [25] M. S. Aslanpour, S. S. Gill, and A. N. Toosi, "Performance evaluation metrics for cloud, fog and edge computing: a review, taxonomy, benchmarks and standards for future research," *Internet of Things*, vol. 12, p. 100273, Dec. 2020, doi: 10.1016/j.iot.2020.100273.
- [26] S. S. Gill, "A manifesto for modern fog and edge computing: vision, new paradigms, opportunities, and future directions," in *EAI/Springer Innovations in Communication and Computing*, 2022, pp. 237–253.
- [27] F. Oliveira, D. G. Costa, F. Assis, and I. Silva, "Internet of intelligent things: a convergence of embedded systems, edge computing and machine learning," *Internet of Things*, vol. 26, p. 101153, Jul. 2024, doi: 10.1016/j.iot.2024.101153.

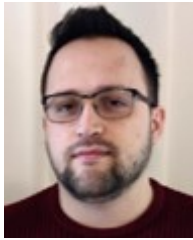
## BIOGRAPHIES OF AUTHORS






**Gaetanino Paolone**    received the Ph.D. degree in Electrical Engineering and Information from the University of L'Aquila, Italy, in 2009. He is currently the CEO of B2B S.r.l. His research interests include Software engineering, software development methodological processes, automatic code generation, AI, and IoT. He can be contacted at email: [g.paolone@b2binformatica.it](mailto:g.paolone@b2binformatica.it).






**Romolo Paesani**    obtained the Software Programming diploma in 2008. He is currently employed as Software Designer, Developer and Analyst at Gruppo SI S.c.a.r.l. His research interests include Software Engineering and automatic code generation. He can be contacted at email: r.paesani@softwareindustriale.it.






**Jacopo Camplone**    received the High School Scientific diploma in 2010. He is currently employed as Software Designer, Developer and Analyst at B2B S.r.l. His research interests include Software Engineering, Software development methodological processes, automatic code generation, and IoT. He can be contacted at email: j.camplone@b2binformatica.it



**Andrea Piazza**    received the bachelor's degree in Automotive Engineering from the Politecnico of Torino, Italy, in 2023. He is currently employed as Software Analyst, Designer and Developer at B2B S.r.l. His research interests include Software engineering and IoT. He can be contacted at email: a.piazza@b2binformatica.it.



**Paolino Di Felice**    is professor of Computer Science since 1999 at the Department of Industrial and Information Engineering and Economics of the University of L'Aquila. He has coauthored about 120 articles appeared in international journals, books, and conference proceedings. He has carried out a consistent activity of technological transfer in collaboration with IT firms. He serves regularly as member of the Editorial board of international journals and as a Program Committee member of conferences. Software engineering and IoT are the two prevalent research topics he is involved in. He can be contacted at email: paolino.difelice@univaq.it.