H2020 Work Programme 2016-2017
H202-IoT-02-2016
Duration: 36 months

CROSS FERTILISATION THROUGH ALIGNMENT, SYNCHRONISATION AND EXCHANGES FOR IoT

H2020 – CREATE-IoT Project

Deliverable 02.02

Reference architecture for federation and cooperation between IoT deployments

Revision: 1.00
Due date: 30-06-2018 (m18)
Actual submission date: 30-06-2018
Lead partner: SINTEF/ATOS

Dissemination level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Summary					
No and name	**D02.02 Reference architecture for federation and cooperation between IoT deployments**				
Status	Released	**Due**	m18	**Date**	30-06-2018
Author(s)	O. Vermesan (SINTEF), R. Bahr (SINTEF), J. Valiño (ATOS), J. Gato (ATOS), M. Serrano (NUIG), T. Teixeira (UNP), M. Álvarez-Díaz (GRAD), D. Chaves (GRAD), Francesco Sottile (ISMB), Agata Tringale (ISMB).				
Editor	O. Vermesan (SINTEF)				
DoW	Reference architecture for federation and cooperation between IoT deployments				
Comments					

Document history			
Rev.	**Date**	**Author**	**Description**
0.00	12-02-2018	ATOS	Table of contents and effort assignment.
0.01	06-04-2018	ATOS	SYNCHRONICITY architecture study added.
0.02	30-04-2018	SINTEF	LSP Architecture analysis (AUTOPILOT).
0.03	07-06-2018	SINTEF	Additional info on LSP Architecture analysis (AUTOPILOT).
0.04	19-06-2018	ATOS	IoF2020 architecture study added.
0.05	22-06-2018	SINTEF	Reference architectures and standards.
0.06	25-06-2018	NUIG	Reference architectures and standards; LSP architecture analysis (ACTIVAGE).
0.07	25-05-2018	SINTEF	Additional info on reference architectures and standards.
0.08	26-06-2018	UNP	Added information on IoF2020.
0.09	26-06-2018	SINTEF	Reference architectures and standards.
0.10	27-06-2018	SINTEF	LSP Architecture analysis (AUTOPILOT IoT architectures).
0.10	28-06-2018	GRAD	Internal review.
0.11	28-06-2018	SINTEF, ISMB	Update and input LSP Architecture analysis (MONICA IoT architectures).
0.12	29-12-2018	ATOS	Review.
0.13	29-12-2018	ISMB	Additional input on LSP Architecture analysis (MONICA IoT architectures).
0.14	29-12-2018	SINTEF	Review comments considered.
1.00	30-06-2018	SINTEF	Final version released.

Table of contents

1. PUBLISHABLE SUMMARY

The current IoT landscape is populated by a wide plethora of single approaches and common proposals in terms of reference architecture. On the former case, usually small or very vertically-oriented use cases adopt solutions just tailored to their domain needs. On the latter case, generic, horizontal and usually use case agnostic proposals are outputted trying to cover all IoT approaches at once. IoT reference architectures are being developed in several organizations, such as IEEE, ISO/IEC JTC 1 Special Working Group 5 (Internet of Things) and Working Group 7 (Sensor Networks), oneM2M, ITU-T, IETF, W3C, etc.

Architecture descriptions are used by the stakeholders that create, utilize and manage complex systems to improve communication and co-operation, enabling them to work in an integrated and coherent way. Architecture frameworks and description languages are created as assets that codify the conventions, common practices and descriptions within different communities and domains of application. In ISO/IEC/IEEE 42010, the conceptualization of a system's architecture, assists the understanding of the system's essence and key properties pertaining to its behaviour, composition and evolution, which can affect concerns such as the feasibility, utility and maintainability of the system. [6].

An architecture description includes one or more architecture views, and the architecture view addresses one or more of the concerns held by the system's stakeholders. The architecture view expresses the architecture of the system-of-interest in accordance with an architecture viewpoint. There are two aspects to a viewpoint: The concerns it frames for stakeholders and the conventions it establishes on views. ISO/IEC/IEEE 42010 does not use phrases such as "business architecture", "physical architecture", and "technical architecture". In the terms of ISO/IEC/IEEE 42010, the architecture of a system is a holistic conception of that system's fundamental properties, best understood via multiple views of that architecture. Therefore, approximate equivalents of the above phrases are "business view", "physical view", and "technical view", respectively. A concern can be framed by more than one viewpoint. A view is governed by its viewpoint: the viewpoint establishes the conventions for constructing, interpreting and analysing the view to address concerns framed by that viewpoint. Viewpoint conventions can include languages, notations, model kinds, design rules, and/or modelling methods, analysis techniques and other operations on views [6].

Even though generic reference architectures might not be optimal for each and every potential use case, it is extremely important, and a current request from most stakeholders in the IoT environment, to convey in a common approach to structure IoT deployments. These generic reference architectures can be eventually fine-tuned for a specific use case or application if needed.

Following this line of thought, the present report aims at, identifying all relevant and most widely adopted reference architectures for IoT deployments. Building on top of this, this deliverable analysis the approach followed by each LSP with respect to architecture. The focus is put on understanding each approach so that, at the end, a core set of architectural guidelines can be outlined as best practices, commonly shared by all LSPs and successfully implemented.

2. OBJECTIVES FOR REFERENCE AND COMPARISON OF IOT ARCHITECTURES IN LSPS

This section is devoted to describing the way CREATE-IoT addresses each LSP to collect the needed data. Once this data is extracted from the appropriate task forces inside each project, the methodology used to compare and extract best practices is also outlined.

2.1 Methodology for Reference Architecture identification

The interaction with LSPs in terms of architectural information is done through Activity Group (AG) 02 (IoT Standardisation, Architecture and Interoperability). This activity group comprises all LSP contacts working directly on the architecture.

The feedback requested was oriented to bring some light with respect to several key items, as described below:

- **Adoption of reference architecture**. Some LSPs are directly embracing a standard with respect to architecture, imposing a common ground for all inherent developments. The other way around, some other LSPs are more open, allowing use case providers to use their own approaches.
- **Architectural focus**. Even for those using reference architecture, most likely they are focusing on just certain parts/modules, as they are the ones critical for their domain. In those cases, it is very important to understand if they are pushing forward the state of the art, enhancing the baseline reference with new developments or particularising the general model for their purposes. On the other hand, there might be some modules not used or not relevant at all.
- **Use case policy**. Some LSPs are very pilot-oriented. This usually means that those pilots are given a certain degree of flexibility to adopt the best option in terms of architecture. This implies several approached to architecture within the same project, shifting the focus on a unified evaluation or result-achievement rather than in a unified architectural approach. On the contrary, having the same model used for all use cases and developments is crucial to other LSPs. In those cases, architecture is put in a central position and efforts from partners are aligned towards homogenising their approaches.
- **Benchmarking**. Going one step further, having a unified reference model will enable comparison and evaluation of similar metrics at least on the core IoT part of all projects. This way, LSPs are asked to make and effort and try to consider a methodology to assure they are compliant with what they need in terms of architecture.

2.2 Comparison of LSP initiatives and way forward

The comparison and guideline proposal expected as an outcome of this report is the result of a well-studied and planned strategy, comprising several stages:

- Early in Year 2, CREATE-IoT organised an Activity Group 02 workshop with all LSPs and oriented to cover reference architecture. All LSPs were invited to present their approaches and answer the questions from the audience.
- Based on the first feedback from the workshop, CREATE-IoT partners analysed the LSP approached and produced a template as described in the previous section. The template was filled in by CREATE-IoT experts connected to Activity Group 02 contacts, so that each LSP is presented using the same structure.

- The templates were analysed to produce the results presented in this report. Special attention is paid to commonalities and benchmarking.
- Once this deliverable is published, the work will continue in Activity Group 02 and evolve further in a set of recommendations and best practices that will form the basis for discussions on pre-normative and standardisation activities across the application domains. The work will further concentrate on promoting the commonalities and best practices to standardization.

3. ANALYSIS OF REFERENCE IOT DEVELOPMENTS AND STANDARDS

3.1 Reference architectures and standards

3.1.1 IoT-A

IoT-A (Internet of Things - Architecture) project [12] proposed an IoT-A Architectural Reference Model (ARM) together with the definition of an initial set of key building blocks [13][24]. Using an experimental paradigm, IoT-A combined top-down reasoning about architectural principles and design guidelines with simulation and prototyping in exploring the technical consequences of architectural design choices. [24][25]. The IoT-A Architecture Reference Model is a set of best practices, guidelines, and a starting point to generate specific IoT architectures [26]. It provides an architectural reference model facilitating interoperability among IoT systems and integration into the service layer.

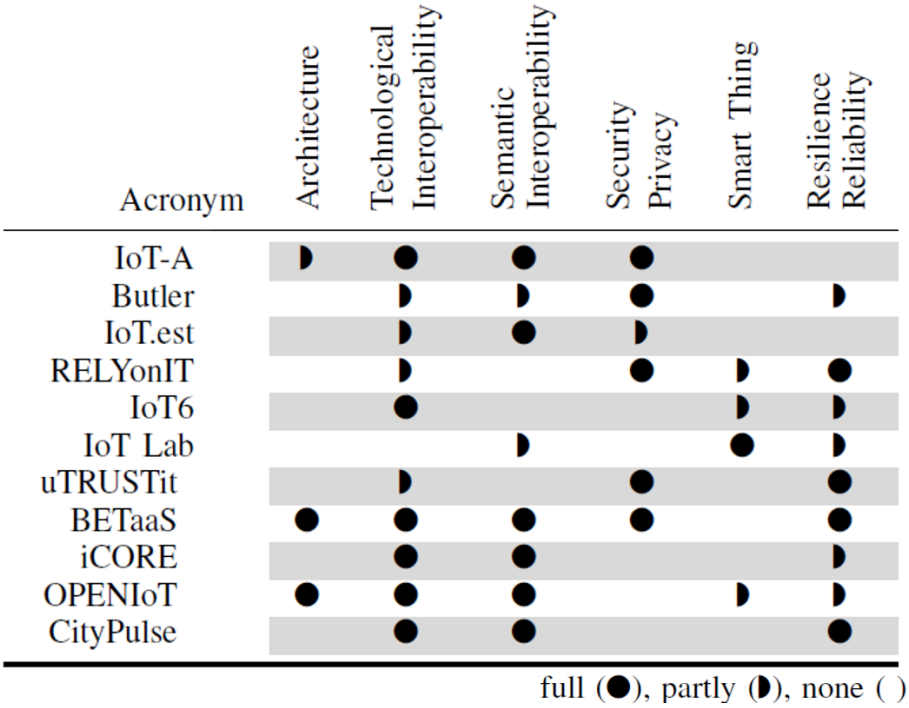


Figure 1: IoT-A compared with other EU funded IoT projects, (Source: E. Gazis, et.al [26]).

IoT-A defines the steps to generate concrete IoT architectures from business goals. The covered topics include the generation of requirements and their transformation into an architecture. IoT-A provides a list of stakeholder unified requirements (UNIs), that can be used to generate concrete requirements for a specific architecture. IoT-A makes use of reference models that introduce major IoT concepts like devices, services, and entities. Their relations and attributes are defined on an abstract level and independent of technologies and use cases. The established relations identify functional groups (FGs) for interacting with instances of the introduced concepts and introduces communication functionalities suitable for heterogeneous IoT settings. Issues addressing trust, security, privacy, interoperability, scalability, process management, and service organization are also included [24][26].

Figure 1, reproduced from [26], shows to what extent different IoT projects (most of them IERC members) address challenges such as technological interoperability; semantic interoperability;

security and privacy; smart things; and resilience and reliability. Figure 2 shows the IoT-A functional architecture.

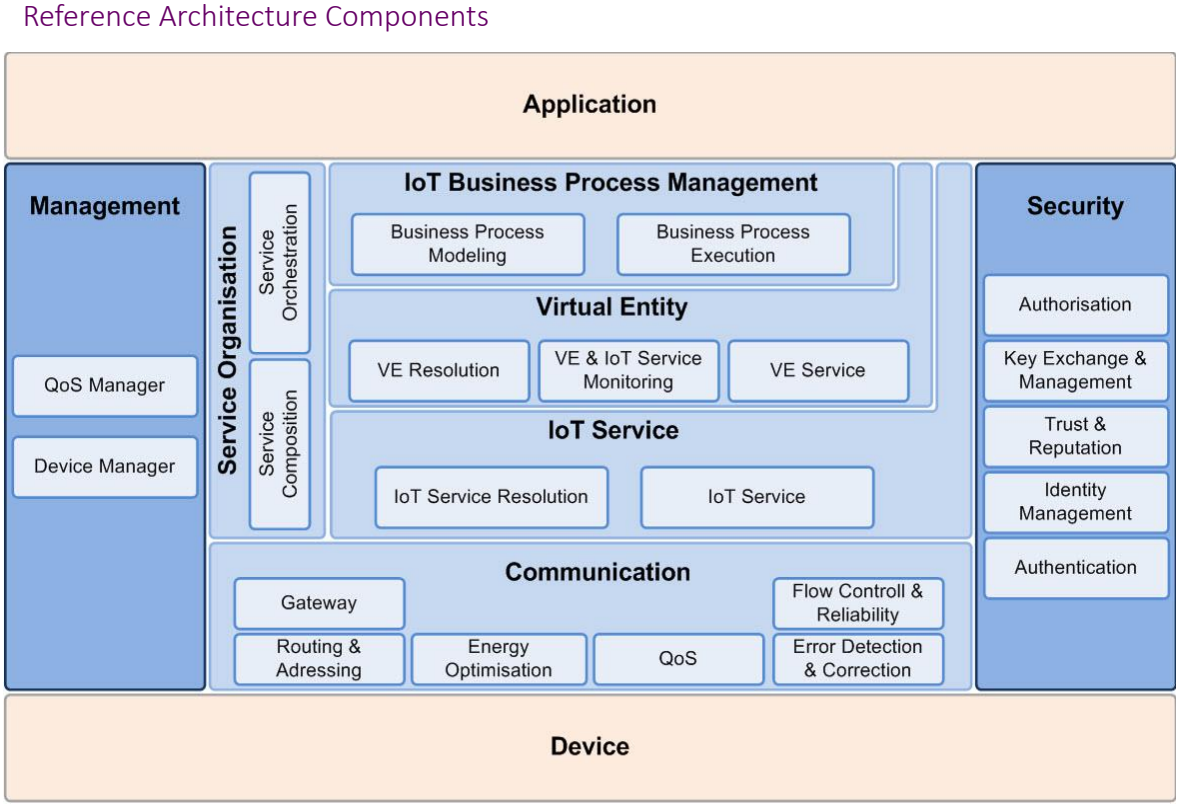


Figure 2: IoT-A Functional Architecture, (Source: IoT-A Project 2013)

3.1.2 ISO/IEC CD 30141

ISO/IEC JTC 1 is working on the Internet of Things Reference Architecture (IoT RA) document that provides a standardized IoT reference architecture using a common vocabulary, reusable designs and industry best practices. It uses a top down approach, beginning with collecting the most important characteristics of IoT, abstracting those into a generic IoT conceptual model, deriving from the conceptual model to a high-level system based reference model and then breaking down from reference model to the five architecture views (functional view, system view, user view, information view and communication view) from different perspectives [7]. The IoT reference architectures (RA) serves the following goals [7]:

- To describe the characteristics of IoT systems;
- To define the domains of the IoT system;
- To describe conceptual model (CM), reference model (RM) of IoT systems; IoT architecture views; and
- To describe interoperability of IoT system's entities.

The IoT RA is also intended to [7]:

- Facilitate the understanding of the overall structure of IoT systems;
- Illustrate and provide understanding of IoT RA from different architectural views;
- Provide a technical reference to enable the international community to understand, discuss, categorize and compare IoT systems; and
- Facilitate the analysis of candidate use cases/applications including data/information flows.

The IoT reference architectures (RA) structure is presented in Figure 3.

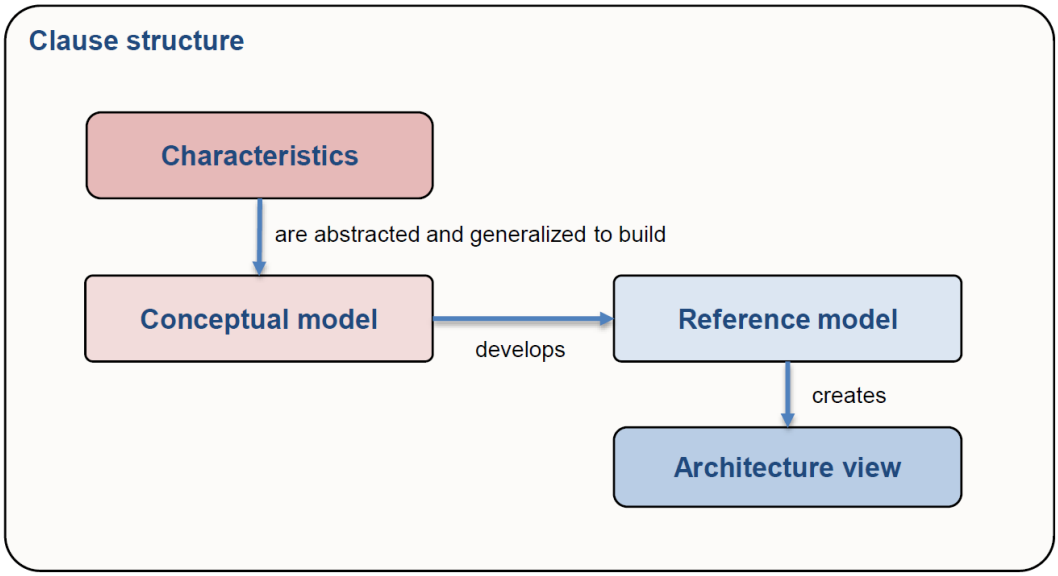


Figure 3: IoT RA structure [7]

The IoT reference architecture enables and supports providing coherent international standards for IoT, while offering a technology-neutral reference point for defining standards for IoT and encourages openness and transparency in the development of a target IoT system architecture and in the implementation of the IoT system.

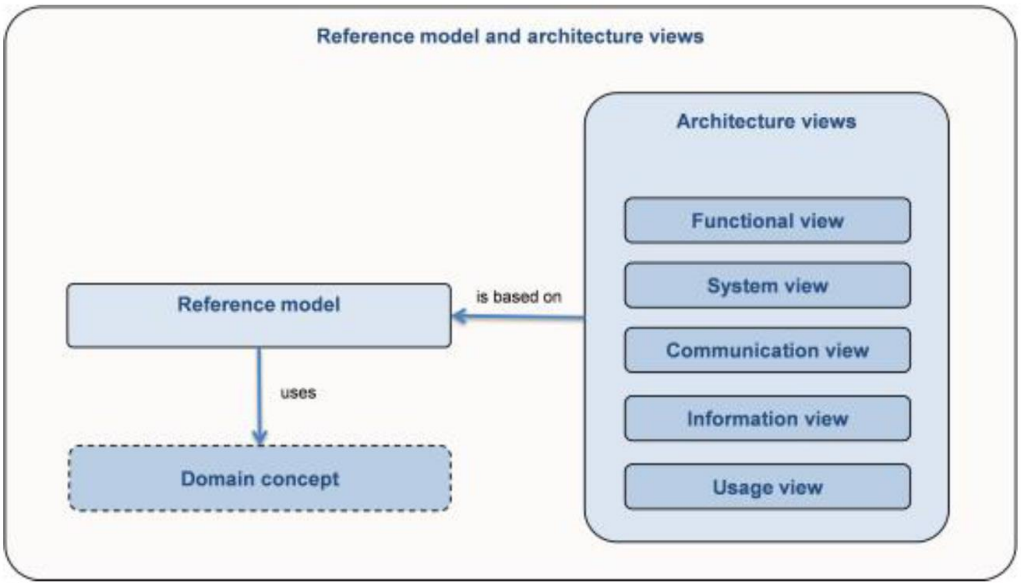


Figure 4: Reference model (RM) and architecture views [7]

ISO/IEC JTC 1 also established a Study Group on Smart Cities including a Smart City Reference Model from an ICT perspective. This reference model is composed of business layer, data layer, sensing layer, security system, and cloud and network resources. The main differentiation point on this model is the inclusion of cloud and network resources, as a vertical to support the architecture. Existing standard from JTC 1 and ITU to support the integration/southbound layers. About interoperability and northbound interactions, an ontology is proposed (PAS 182 Smart city concept model [34]).

The ISO/IEC 30141 Internet of Things Architecture (IoT RA) is still not an ISO international standard, but a committee draft (CD) under development which is distributed for review and comment and subjected to changes. However, the prevailing draft version is our basis for the

analysis [27]. The IoT RA provides IoT system characteristics and domains; IoT system conceptual and reference models; different views on reference architectures; and interoperability of IoT system entities. The IoT RA also provides rules and guidelines for developing IoT system architectures.

The architecture hierarchy is well described from conceptual model (CM) through the entity-based and domain-based reference model (RM) to several different views of the reference architecture (RA) as illustrated in Figure 5 [27]. The IoT RA functional view and decomposition of the functional components are shown in Figure 6 [27].

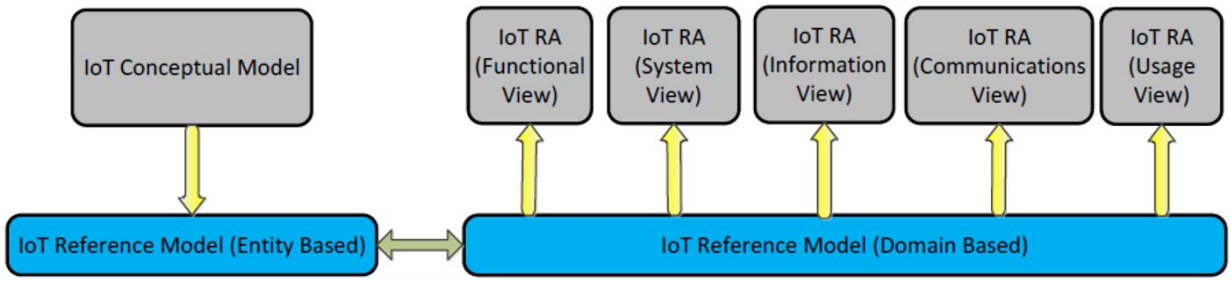


Figure 5: Relation between IoT CM, RM and RA, (Source: ISO/IEC CD 30141[27]).

The CM provides a common structure and definitions for describing the entities within IoT systems. The concepts and relationships are described in a generic, abstract and simple way. Unified modelling language (UML) is used to clarifying the fundamentals of the IoT systems.

The RM are based on functional, system, communication, information, and usage architecture views, and is an abstract framework for understanding significant relationships among the entities of an environment. RM is not directly tied to any standards or technologies but is a tool for developing consistent standards or specifications supporting that environment and provides common semantics that can used unambiguously across and between different implementations.

The RA provide common features, vocabulary, and requirements, together with supporting artefacts, which are the description of the major architecture components providing guidelines and constraints for solution architectures that can be defined from different viewpoints and at many different levels of detail and abstraction.

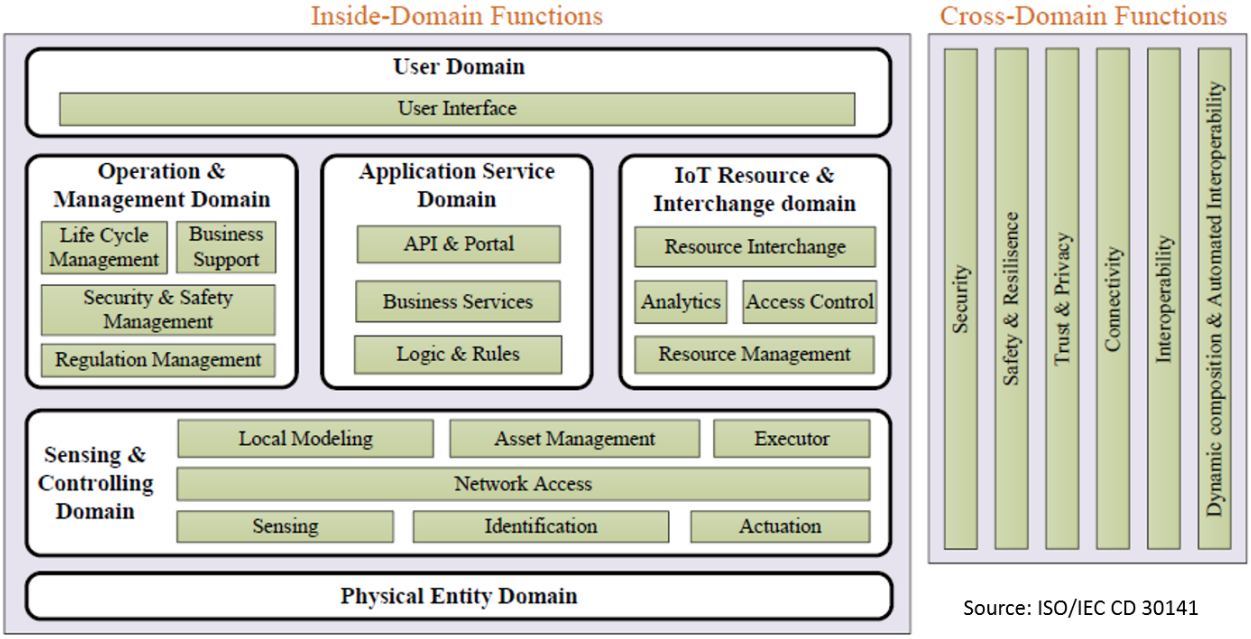


Figure 6: IoT RA functional view, (Source: ISO/IEC CD 30141 [27]).

The main characteristics of IoT systems are described, including relevance and examples. Functions based on these characteristics can be implemented in IoT systems according to services and operations. The characteristics are grouped as follows [27]:

- **IoT system** - auto configuration, function and management capabilities separation; highly distributed systems, network communication, network management and operation, real-time capability, self-description, and service subscription.
- **IoT service** - timeliness, content and context awareness.
- **IoT components** - composability, discoverability, modularity, connectivity, share ability, and unique identification.
- **Compatibility** - well defined components and legacy support.
- **Usability** - manageability and flexibility.
- **Robustness** - reliability, resilience and accuracy.
- **Security** - safety, confidentiality, integrity, availability and protection of personally indefinable information (PII).
- **Other characteristics** - scalability, data volume/velocity/veracity/variability/variety (Data 5Vs), heterogeneity, trustworthiness, and regulation compliance.

Examples mentioned regarding IoT system characteristics and network communication are IEEE 802.15.4 and IEEE 802.11 in communication protocols on physical and data link layers, and 6LowPAN for data transmission, although examples exist on higher layers. The data are routed between the proximity network and the wide area network as necessary.

3.1.3 IEEE P2413

The work covered by IEEE P2413 defines an architectural framework for the IoT, including descriptions of various IoT domains, definitions of IoT domain abstractions, and identification of commonalities between different IoT domains.

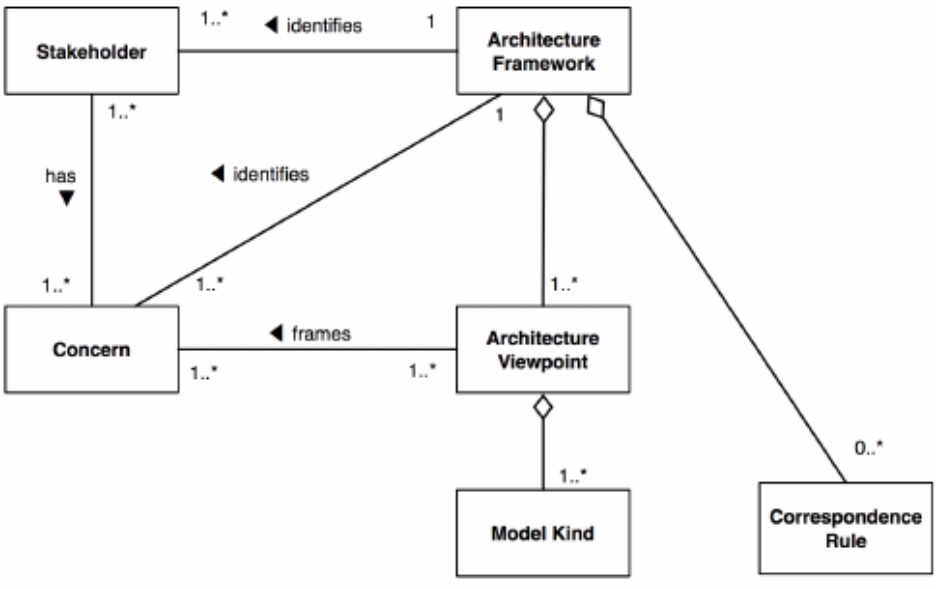


Figure 7: Conceptual model of an architecture framework [5]

The architectural framework for IoT provides a reference model that defines relationships among various IoT verticals (e.g., transportation, healthcare, etc.) and common architecture elements. It also provides a blueprint for data abstraction and the quality "quadruple" trust that includes protection, security, privacy, and safety." Furthermore, this standard provides a reference architecture that builds upon the reference model. The reference architecture covers the definition of basic architectural building blocks and their ability to be integrated into multi-tiered systems. The reference architecture also addresses how to document and, if strived for, mitigate architecture

divergence. This standard leverage existing applicable standards and identifies planned or ongoing projects with a similar or overlapping scope [5].

The IEEE P2413 is a top down approach that follows the recommendations for architecture descriptions defined in ISO/IEC/IEEE 42010 Systems and software engineering - Architecture description [6]. ISO/IEC/IEEE 42010 provides a core ontology for the description of architectures, specifies provisions that enforce desired properties of architecture frameworks, is used to establish a coherent practice for developing architecture frameworks and can be used to assess conformance of an architecture framework.

The IEEE P2413 architecture framework including the abstract IoT domain and the industrial domains that drive the rationalisation of common stakeholders and shared concers is presented in Figure 8.

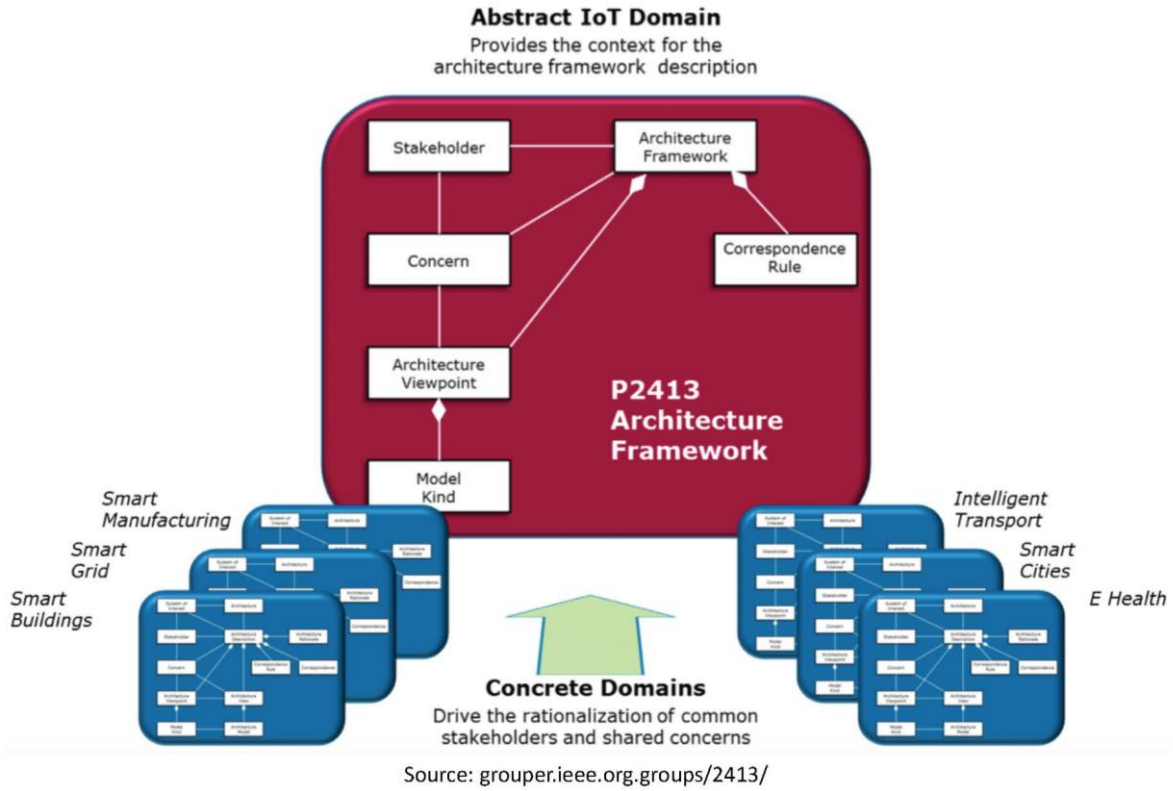
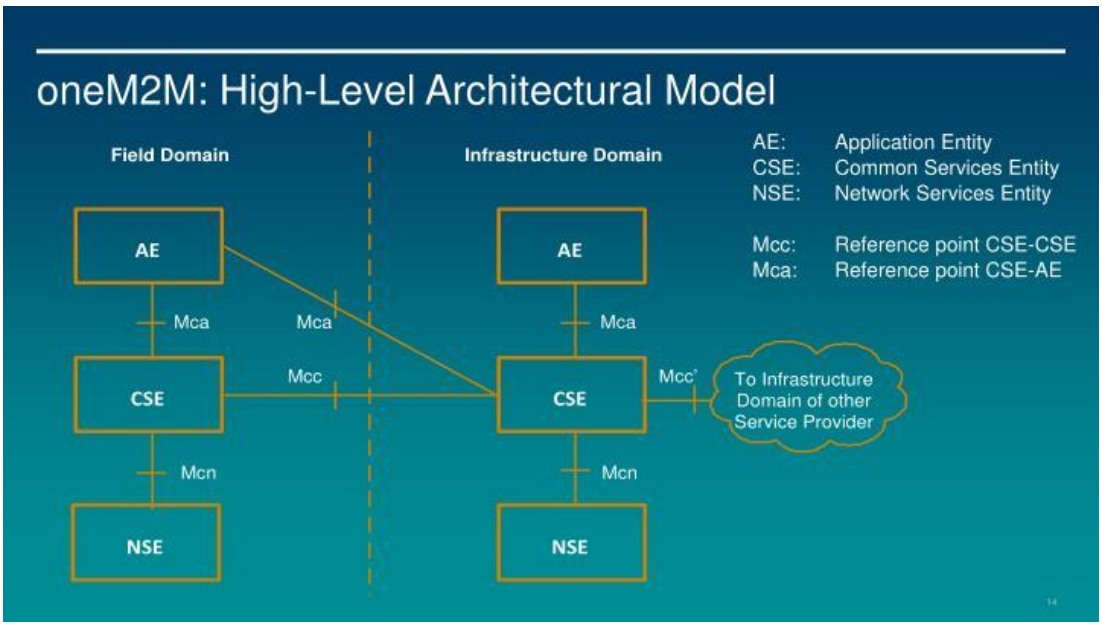


Figure 8: IEEE P2413 Architecture

3.1.4 OneM2M

oneM2M [23] is a partnership project between the leading telecom standardization organizations in Asia, America and Europe. oneM2M is developing specifications for the service layer for machine-to-machine communication and the IoT [2]. oneM2M aims to provide common services layer to IoT applications and devices of different service domain/verticals.

Before the standardization like oneM2M, different systems, consisting of applications, devices and networks, needed to be developed per service and modified for replication for other usages. With this, the IoT ecosystem could not easily have scale of economy but have fragmented markets. The oneM2M common services layer, in other words middleware, provides common service functions to applications and devices in form of APIs. By providing the common services layer, different vendors and service domains can use the same APIs and then services and devices can increase interoperability drastically compared to the situation before.



Source: oneM2M 2016

Figure 9: oneM2M High Level Architecture

oneM2M High Level Architecture is depicted in Figure 9. In the oneM2M functional architecture two basic types of entities are defined. One is an **AE** (short for Application Entity) and the other is a **CSE** (short for Common Services Entity). An **IN-CSE** (short for Infrastructure Node CSE) is hosted in the cloud by the oneM2M Service Provider and a **MN-CSE** (short for Middle Node CSE) is hosted on the Home Gateway [8].

The **Mca** reference point defined by oneM2M is used to interface an AE and CSE, for example a lighting AE and a home gateway (CSE). The **Mcc** reference point defined by oneM2M is used to interface between CSEs, for example between a home gateway and a cloud service platform.

3.1.5 ITU-T FG-SSC (ITU-T Y.4414/H.623)

Based on the work of the FG-SSC (Smart Sustainable Cities) [51], key desirable features for smart sustainable cities along with defined key performance indicators to monitor smart city transitions, have been elaborated in the online book "Shaping smarter and more sustainable cities: Striving for sustainable development goals" [28][29].

This work stems from the study by the ITU-T Group 5 on Smart Sustainable Cities aiming for a smart-city open platform. It consists in four logical layers (application, data, network and sensing) emphasizing a physical perspective. There are no reference implementations, but it supports two standards for open data management (ISO 11179 for data modelling [49], ISO 15000 webXML for web services [50]).

The work on smart sustainable cities is continued by the ITU-T Study Group SG20 on “Internet of things (IoT) and smart cities and communities (SC&C)” [52] established in 2015, which provides a platform to influence the development of international IoT standards and their application as part of urban development plans [28][30]. SG20 provides developers of IoT standards, with the opportunity to target their standardizations efforts towards specific applications and various urban parameters, thereby responding to the requirements of standards implementers including city administrations, energy and water utilities, healthcare providers, and transportation authorities [28].

The recommendations Y.4414/H.623 are maintained by SG20 (originally SG16) and define a web of things (WoT) service architecture that can encompass service discovery, accessibility, sharing and mash-up for IoT devices [33]. It includes an overview of WoT services, the functional

architecture of WoT services and functions, and the architecture supports accessibility and reusability across IoT resources. It supports portability across heterogeneous network environments for seamless integration with information interaction and exchange over physical IoT devices. The functional architecture of WoT services is illustrated in Figure 10 [33], where S is service, FE is functional entity and T2R is things to resource.

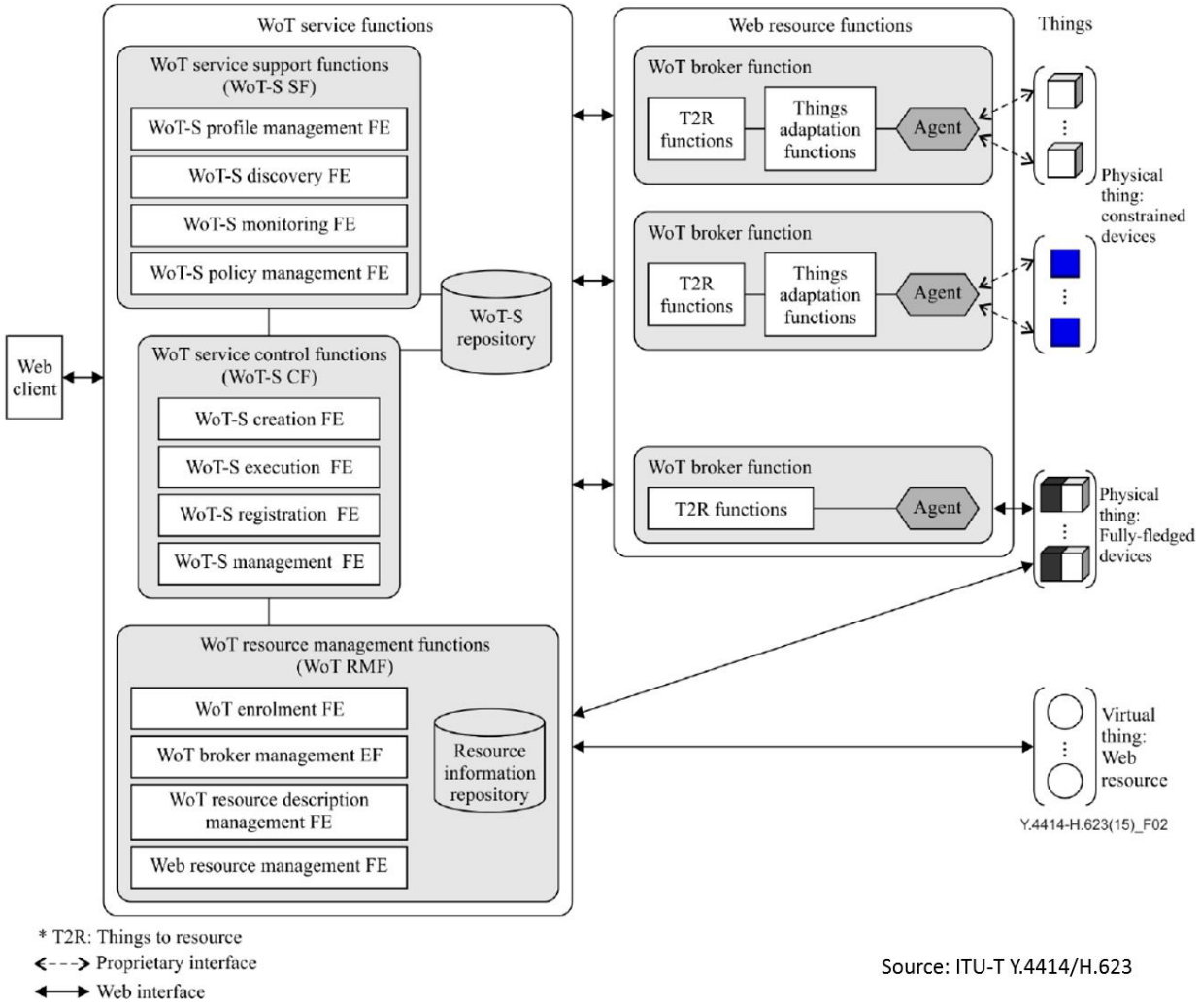


Figure 10: ITU-T Y.4414/H.623 Functional architecture of WoT service [33]

The WoT service functions provide the services to the clients, and manage the overall behaviours related to the WoT service. If a client requests a service, the WoT service functions analyse the request, and discover/provide the services for the client. The WoT service functions have three sub-functions [33]:

- Service support functions (WoT-S SF) - manages the overall behaviours of the service, and is responsible for providing service profile management, service discovery, service monitoring, QoS management, updates, access control and policy management of the WoT service.
- Service control functions (WoT-S CF) - control the service which are registered at the service providers and is responsible for executing/creating the WoT and mash-up services.
- Resource management functions (WoT-S RMF) - contain resource (things) information which is used by and registered at the WoT service functions.

3.1.6 ITU-T SG13 Y.2060

The study group SG13 Future networks focuses on IMT-2020 cloud computing and trusted network infrastructure [31]. The recommendation ITU-T Y.2060 provides an overview of the IoT;

clarifies the concept and scope of the IoT, identifies the fundamental characteristics and high-level requirements of the IoT and describes the IoT reference model as illustrated in Figure 11 [32].

The reference architecture consists in four layers (application, service & application support, network and device), complemented with two verticals (management and security). These two verticals cover generic functionalities about fault tolerance, configuration, authentication, authorization and similar [32].

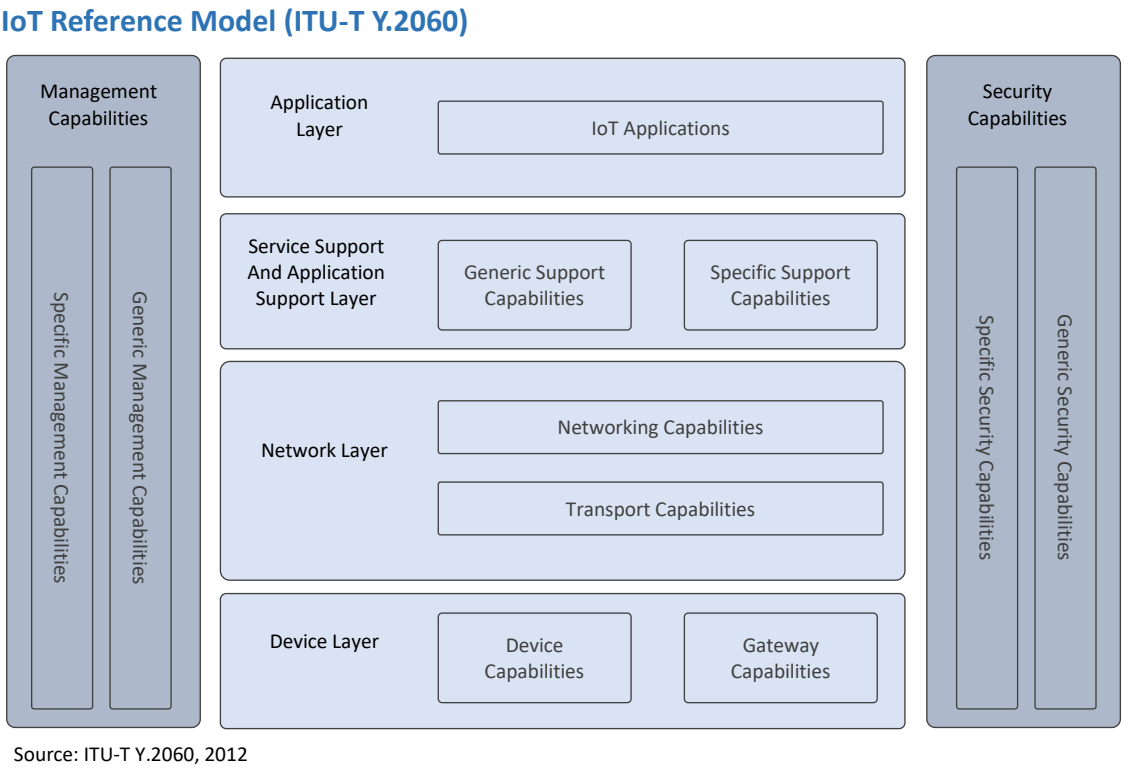


Figure 11: ITU-T Y.2060 IoT reference Model Architecture [32]

At the device layer, the gateway capabilities support devices connected through different kinds of wired or wireless technologies, such as a CAN bus, ZigBee, Bluetooth or Wi-Fi. At the network layer, the gateway capabilities may communicate through various technologies, such as the PSTN, 2G or 3G networks, LTE networks, Ethernet or digital subscriber lines (DSL) [32].

Regarding protocol conversion, there are situations where gateway capabilities are needed. For example, when communications at the device layer use different device layer protocols such as ZigBee and Bluetooth, or when communications involving both the device layer and network layer use different protocols such as ZigBee at the device layer and a 3G at the network layer [32].

The network layer consists of networking capabilities (access control functions, mobility management, authentication, authorization and accounting (AAA)) and transport capabilities (connectivity for the transport of service and application data information, as well as IoT control and management information).

The immediately higher layer provides generic service and application support capabilities (data processing, data storage) and specific support capabilities (tailored to provide different support functions to IoT applications).

IoT applications are contained in the higher layer. Additionally, the model contains management and security capabilities that can be generic (for example: device management, local network topology management, application data confidentiality, privacy protection) or application-specific.

3.1.7 ETSI CIM (FIWARE)

FIWARE is an open-source software platform which can be assembled and deployed to develop smart and secure solutions in a wide range of application domains, exploiting cutting-edge technologies like IoT, Cloud architectures or Big Data [ref]. Initially developed as part of a European public-private partnership program, it is promoted from 2016 by the FIWARE foundation [14].

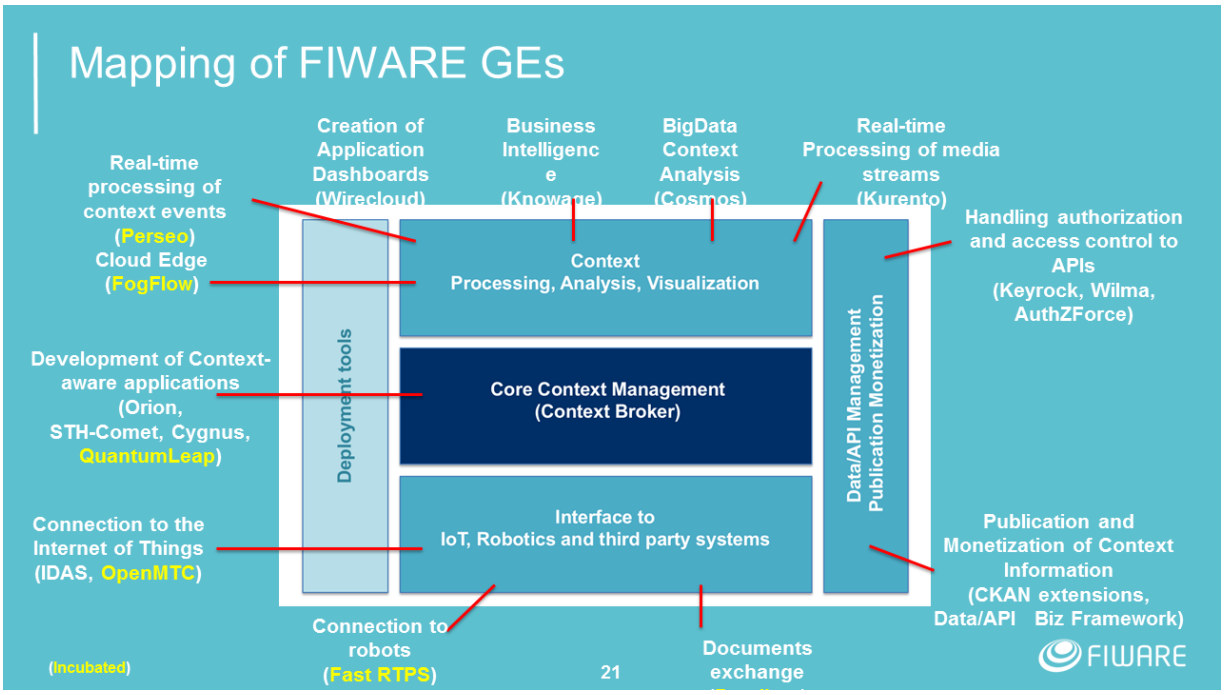


Figure 12: Overview of FIWARE architecture and GEs

(Source: FIWARE Overview, FIWARE Global Summit 2018)

One of its main value propositions is the specification of ETSI NGSI-LD, a standard Application Programming Interface (API) to manage Context Information Management at large scale [15].

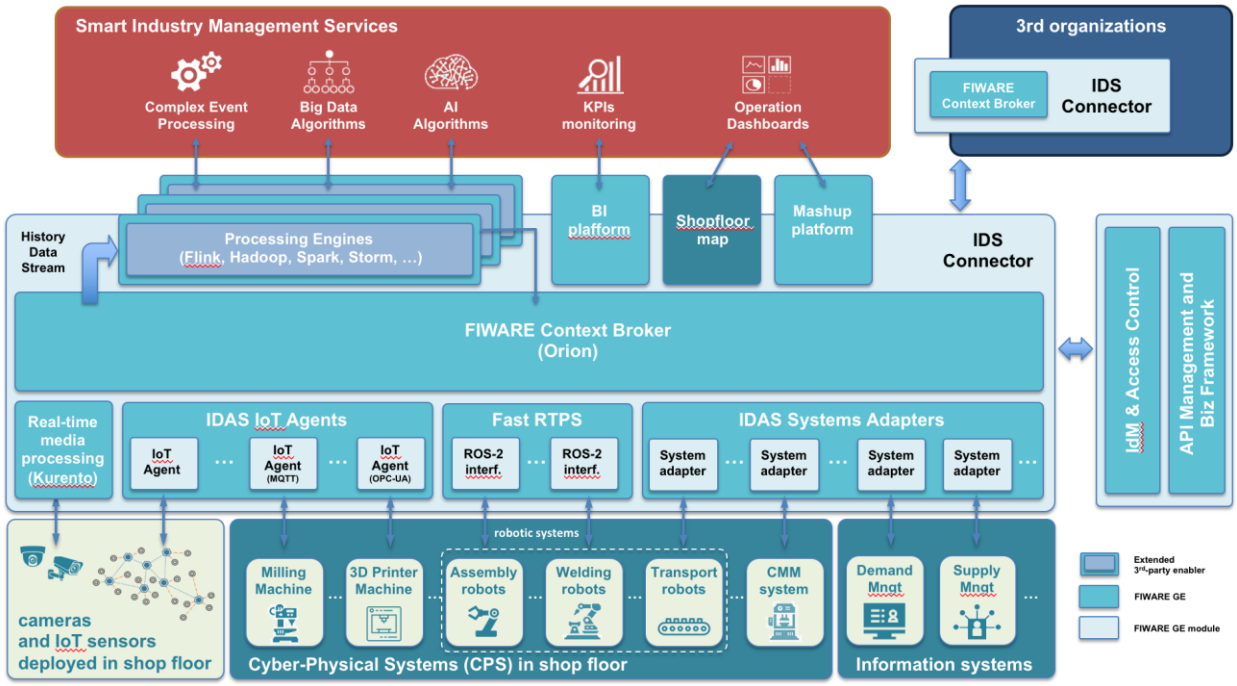


Figure 13: FIWARE reference architecture in Smart Factory or Industry 4.0

(Source: FIWARE Overview, FIWARE Global Summit 2018)

The adoption of this API in order to get access to context data allows breaking silos and vendor-locks and abstracts the complexity and low-level details of the multiple IoT protocols that may be part of the same system. An open-source reference implementation of NGSI-LD standard is provided with FIWARE Context Broker component [16].

In addition, FIWARE defines the concept of Generic Enablers (GEs): a set of open-source specifications and reference implementations for additional components that complement FIWARE Context Broker functionalities [17]. Figure 12 below depicts a high-level overview of FIWARE architecture and the mapping of currently existing GEs,

As it can be seen, it flexible philosophy enables customized solutions to be tackling the specific requirements and needs of very different sectors. In this sense, specialized reference architectures are proposed in verticals like smart cities, smart farm management or smart factories. The latest example is presented in Figure 13.

To complement NGSI-LD standard, the GEs and the reference architectures, FIWARE also provides a collection of harmonized data models which have been specified exploiting the expertise and know-how acquired in multiple pilots, research projects and commercial initiatives [18].

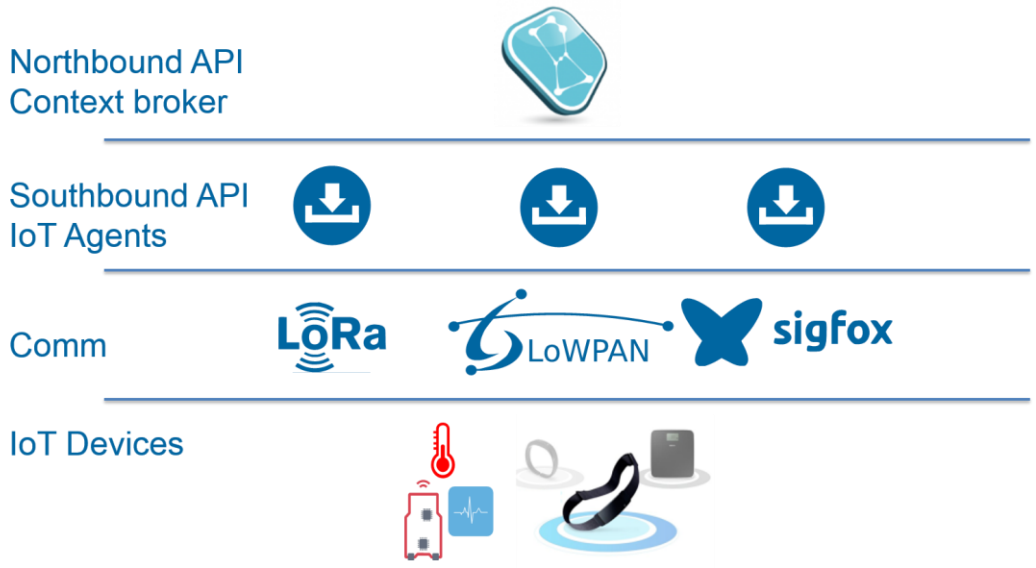


Figure 14: Role of IoT Agents within FIWARE ecosystem

(Source: Connecting FIWARE to IoT, FIWARE Global Summit 2018)

From the IoT device perspective, FIWARE does not impose any restriction regarding the communication protocol. It leverages the concept of the IoT Agents which support the interconection of devices by translating the corresponding IoT protocol to NGSI.

3.1.8 OASC

The Open & Agile Smart Cities (OASC)[36] initiative is a city-driven non-profit organization founded in January 2015. The idea behind this institutional movement is to overcome the lack of standards and offer a smooth digital transition for cities and communities, based on their needs.

Therefore, the main goal, going a step beyond the “granular” concept of smart-cities, is to lead to a fully-fledged smart city “data + services” marketplace.

As a matter of fact, at the time of writing this deliverable, 117 cities coming from up to 24 different countries are active members of the OASC federation. Said in other words, OASC community endeavours to give rise to the Minimal Interoperability Mechanisms (MIM) required for the creation of a smart-city market.

Shifting to a more technical plane, OASC focuses on off-the-shelf and open data platforms and solutions that have been extensively assessed before, instead of coming up with a brand new IoT infrastructure from scratch. With the clear objective of guaranteeing interoperability among cities, OASC opted for a so-called "driven-by-implementation" approach, where communities and developers can "co-create" and tailor their own services.

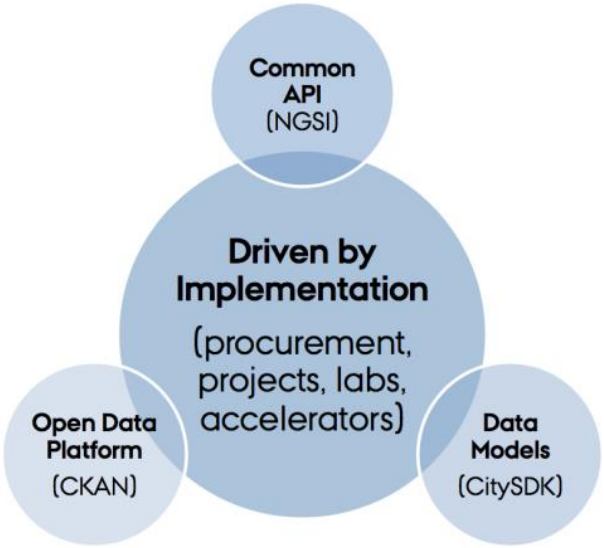


Figure 15: OASC platform "driven-by-implementation" principle

Namely, this principle relies on three main domains or areas, as shown in Figure 15.

- Common API. With a tight binding to the outcomes of FIWARE, OASC relies on their well-known NGSI APIs, which do not only provide the means publish and gather (request/response) context information, but also as a lightweight asynchronous subscription-based framework. Moreover, they also provide mechanisms to link to well-known persistence infrastructures, thus opening the door to future data analysis activities.
- Data Models. To overcome the lack of (official) data models for NGSI interfaces, OASC provides a first approach, gathering the results yielded in the CitySDK project[37]. Nonetheless, these models are prone to be modified/updated upon the feedback of current or forthcoming platforms, meaning that e.g. a city with new datasets is perfectly able to request the addition of these new elements. This process brings about a twofold effect: on the one hand, data models are contrasted against others coming from either apps or other cities; moreover, the community itself gets intrinsically involved in the curation process.
- Open Data platform. When it comes to deal with the storage/publication of data, OASC relies on CKAN[18] as the open-source open-data platform, as it has been defined and integrated as a component within the FIWARE Reference architecture[14].

As can be seen, this initial line-up of solutions was chosen so that OASC could achieve a full compliance level with the FIWARE Reference Architecture. This way, every platform that had adopted FIWARE in the past would become straightforwardly part of the OASC initiative.

3.1.9 W3C

The IoT has huge potential but is highly fragmented with myriad technologies and a lack of interoperability for devices and platforms.

This introduces friction and is holding back opportunities for open markets of services. W3C is addressing this through a high-level abstraction layer using RDF and Linked Data to describe things and their relationships to the context in which they are situated.

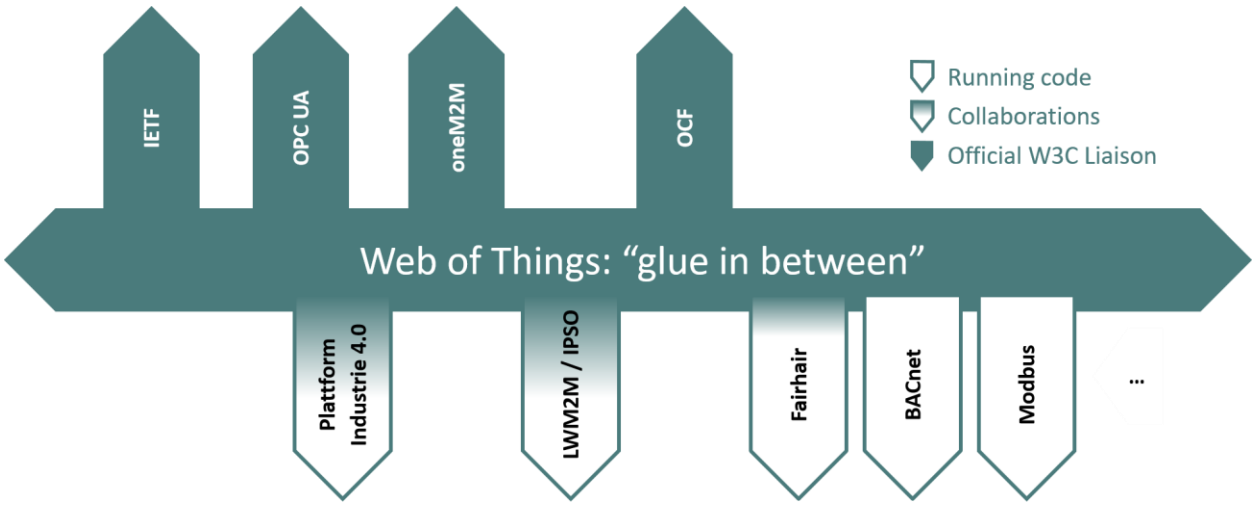


Figure 16: Web of Things as Neutral, Horizontal, Cross-domain Solution

Things are exposed to applications as objects with properties, actions and events, independently of the underlying communication technologies. Every "thing" has a URI that acts as its name, and which can be dereferenced to a Linked Data description of that thing, including the interaction model and data types for the associated properties, actions and events.

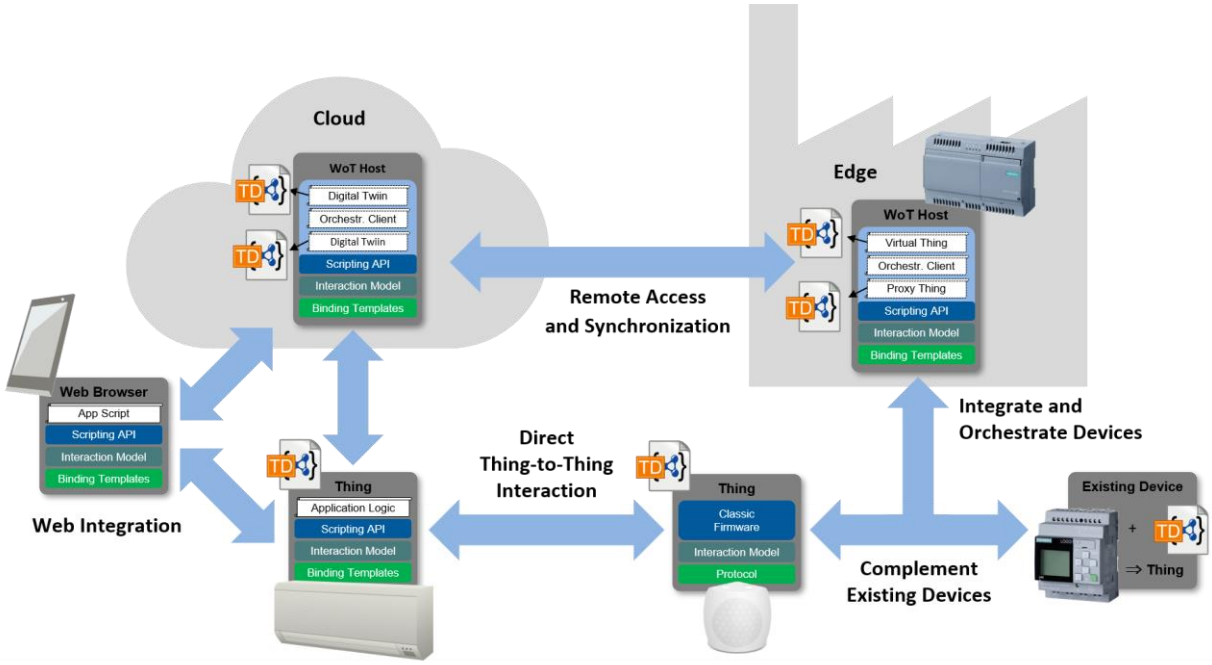


Figure 17: Things and cloud instead of things to cloud

W3C’s Web of Things Working Group [46] is developing standards for the Linked Data vocabulary for the object model of things, a scripting API, and a set of declarative binding templates for common protocols. Open source implementations include the Eclipse ThingWeb [47].

The Web of Things seeks to enable open markets for providers and consumers of services. In practice, IoT technologies are only relevant to the network edge, and a much smaller set of Web protocols will be sufficient for connecting providers and consumers of services across the Internet. Open standards for semantic vocabularies are needed to support discovery based on the kinds of things and their context, composition of services and adaptation to variations in capabilities for similar devices from different vendors.

Additional vocabularies are needed for data licenses, terms & conditions, security and privacy. Shared ledgers based upon blockchains are expected to play an important role in respect to smart contracts, and regulatory requirements for privacy and safety.

The Web of Thing is positioned as a means to bridge different suites of standards, as it is unlikely that any one of these will fulfil the disparate needs of different application domains. The Web of Things can be used flexibly across the network edge, cloud and in between (aka “fog"). Further details of the architecture are given in [48].

3.1.10 AIOTI High-Level Architecture (HLA)

The HLA primarily introduces a domain model, which describes entities in the IoT domain and the relationships between them, and a functional model, which describes functions and interfaces (interactions) within the IoT domain. The HLA Domain Model, is derived from the IoT-A Domain Model. The domain model captures the main concepts and relationships in the domain at the highest level. The naming and identification of these concepts and relationships provide a common lexicon for the domain and are foundational for all other models and taxonomies.

In this model, a *User* (human or otherwise) interacts with a physical entity, a *Thing*. The interaction is mediated by an *IoT Service* which is associated with a *Virtual Entity*, a digital representation of the physical entity. The *IoT Service* then interacts with the *Thing* via an *IoT Device* which exposes the capabilities of the actual physical entity.

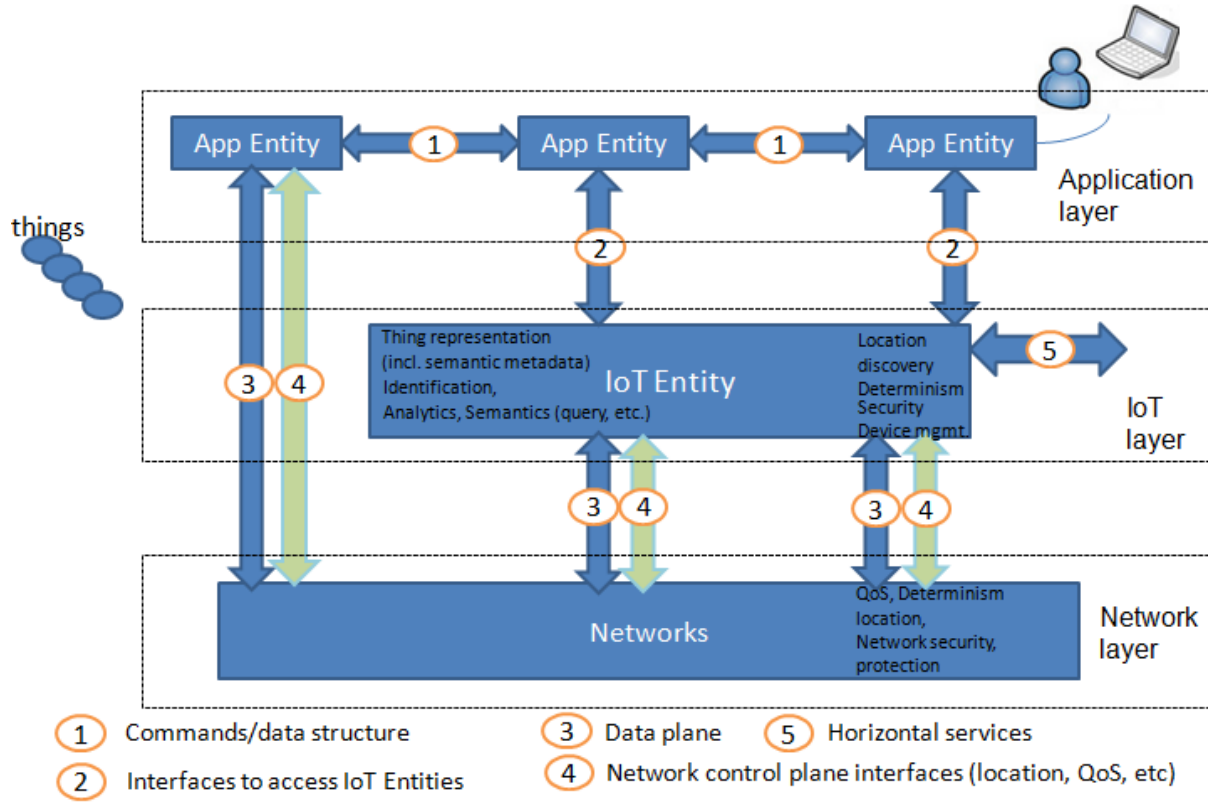


Figure 18: Functional Model of AIOTI HLA

The AIOTI functional model describes functions and interfaces between functions of the IoT system. Functions do not mandate any specific implementation or deployment. Figure 18 shows the HLA functional model. Functions depicted in Figure 18 are:

- App Entity: is an entity in the application layer that implements IoT application logic. An App Entity can reside in devices, gateways or servers. A centralized approach shall not be assumed. Examples of App Entities include a fleet tracking application entity, a remote blood sugar monitoring application entity, etc.

- IoT Entity: is an entity in the IoT layer that exposes IoT functions to App Entities via the interface 2 or to other IoT entities via interface 5. Typical examples of IoT functions include: data storage, data sharing, subscription and notification, firmware upgrade of a device, access right management, location, analytics, semantic discovery etc. An IoT Entity makes use of the underlying Networks' data plane interfaces to send or receive data via interface 3. Additionally, interface 4 could be used to access control plane network services such as location or device triggering.
- Networks: may be realized via different network technologies (PAN, LAN, WAN, etc.) and consist of different interconnected administrative network domains. The Internet Protocol typically provides interconnections between heterogeneous networks. Depending on the App Entities needs, the network may offer best effort data forwarding or a premium service with QoS guarantees including deterministic guarantees.

4. LSP ARCHITECTURE ANALYSIS

The IoT European Large-Scale Pilots Programme includes projects that address the IoT applications based on European relevance, technology readiness and socio-economic interest in Europe. It utilises a rich portfolio of technologies and tools that have been developed and demonstrated in reduced and controlled environments.

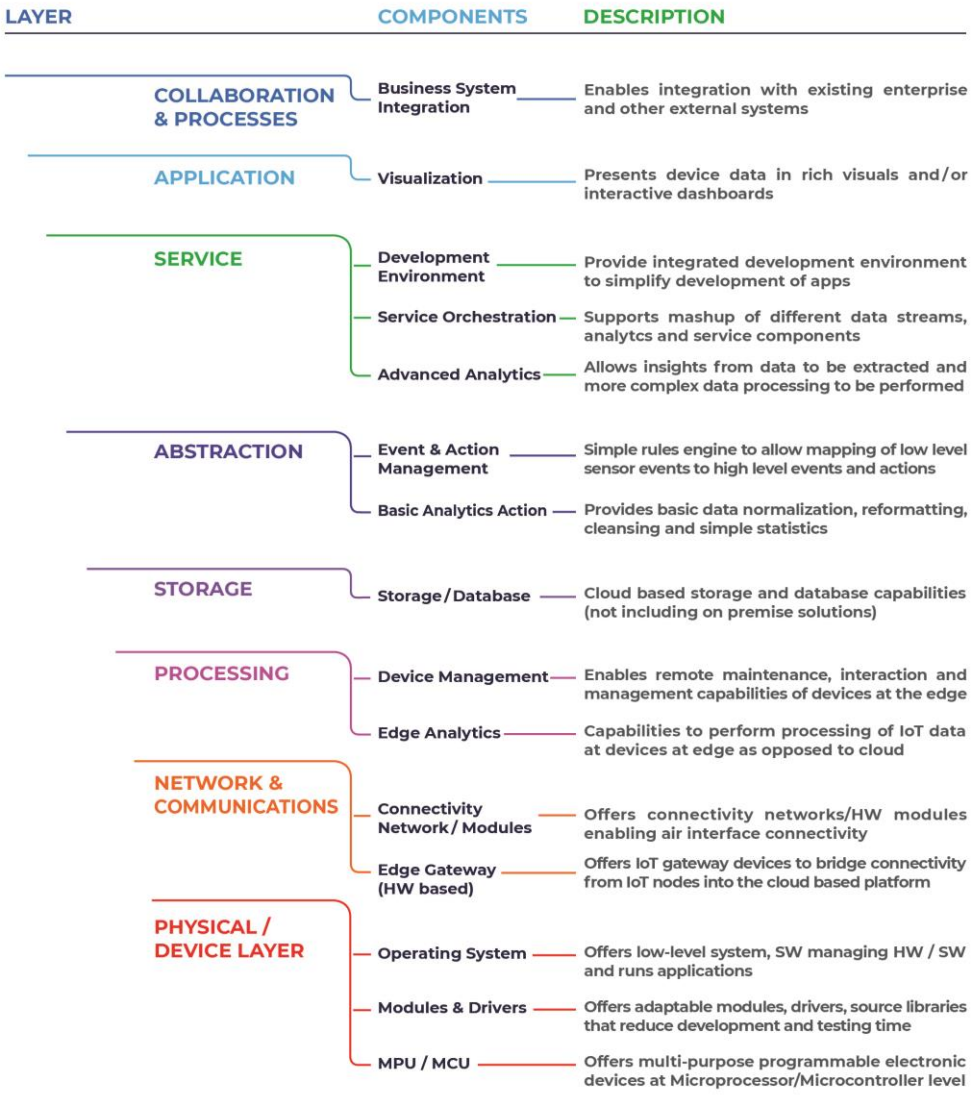


Figure 19: IoT architecture layers

The IoT European Large-Scale Pilots Programme addresses the mapping of pilot architecture approaches with validated IoT reference architectures to enable interoperability across use cases by offering either common or interoperable object connectivity/functionality/intelligence approaches on various levels, such as protocols, data formats that use IoT related enablers and services.

The pilot projects address those elements that provide the basis for interoperability with related fields outside them, especially for key aspects, such as object identification/naming, service publication characteristics, searches and semantic properties.

The pilot architecture approaches are based on the IoT layered architecture shown in Figure 19, which illustrates the functions included in every layer. It also illustrates the components in the IoT architecture's layers that support the feature analysis of different IoT platforms, which make the comparison of solutions easier.

4.1 AUTOPILOT

The AUTOPILOT reference architecture is developed to leverage autonomous driving and innovative mobility services based on open IoT platforms [1]. The AUTOPILOT architecture is used as a common framework to realise IoT-based automated driving use cases. IoT components are deployed at several permanent pilot sites across Europe. The reference architecture consists of a set of services that have various capabilities including processing, communication, resource management, context management, and security. This architecture is composed of an applications layer, an IoT layer, a network layer and some external services (e.g. web services) as well as IoT devices layer. The IoT devices include smart phones, cameras, roadside units, traffic lights and signs, autonomous vehicles, and drones. Within the IoT layer, open IoT platforms based on interoperable and federated models support IoT applications and services of the different use cases and the European pilot sites.

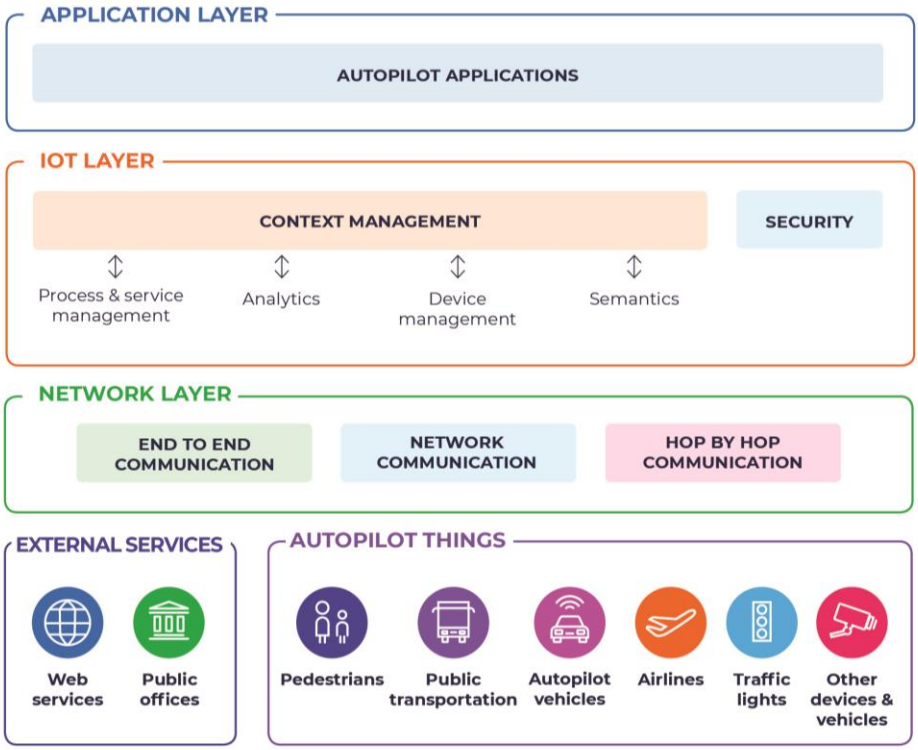


Figure 20: AUTOPILOT architecture simplified.

4.1.1 Reference architecture

Positioning of AUTOPILOT in the IoT landscape cover the following architecture layers and components: Application layer (visualization); Service layer (service orchestration and advanced analytics); Abstraction layer (event/action management and basic analytics action); Storage layer (storage/database); Processing layer (edge analytics); Networking and communication layer (connectivity network/modules and HW based edge gateway); and Physical/device layer (operating system, modules/drivers, and MPU/MCU).

The oneM2M standard (see section 3.1.4) defines two mechanisms to connect oneM2M and non-oneM2M devices and their applications into the IoT platform [1][2]. The native oneM2M devices (or applications) can interact directly with the oneM2M platform using the reference point for M2M communication with the Application Entity (AE), that is the Mca interface; while the non-oneM2M devices (or applications) need a dedicated Interworking Proxy Entity (IPE) developed for this purpose. The IPE provides interworking between the oneM2M platform and the specific IoT device (or application) technologies or protocols.

FIWARE (see section 3.1.7) focuses on a common data model and powerful interfaces for searching IoT device information and use the Open Mobile Alliance (OMA) Next Generation Service Interface (NGSI) data model as the common information model of IoT-based systems and the protocol for communication [3].

NGSI-9 and NGSI-10 are HTTP-based protocols which support JSON and XML formats for data. The NGSI9 protocol is used to manage the availability of context entity.

A system component can register the availability of context information, and later the other system component can issue either discover or subscribe messages to find out the registered new context information [3][9], while the NGSI10 protocol is used to enable the context data transfer between data producers and data consumers, and has query, update, subscribe and notify context operations for providing context values [3][10].

4.1.2 Architectural focus

The integration of autonomous vehicle technologies with IoT technologies requires the cross-fertilisation with other IoT applications in various domains. The project foster links with communities of users and providers in IoT and autonomous transport focus areas, as well as with the other LSPs covering various application domains.

In AUTOPILOT, an IoT open vehicle platform and architecture will be developed and take into account the existing and forthcoming standards as well as open source and vendor solutions. The platform and architecture are based on the AIOTI high level architecture functional model [11], which describes the functions and interfaces between the three functional layers representing the main required entities of an IoT architecture, the application layer, the IoT layer and the network layer, as well as the interfaces between the layers.

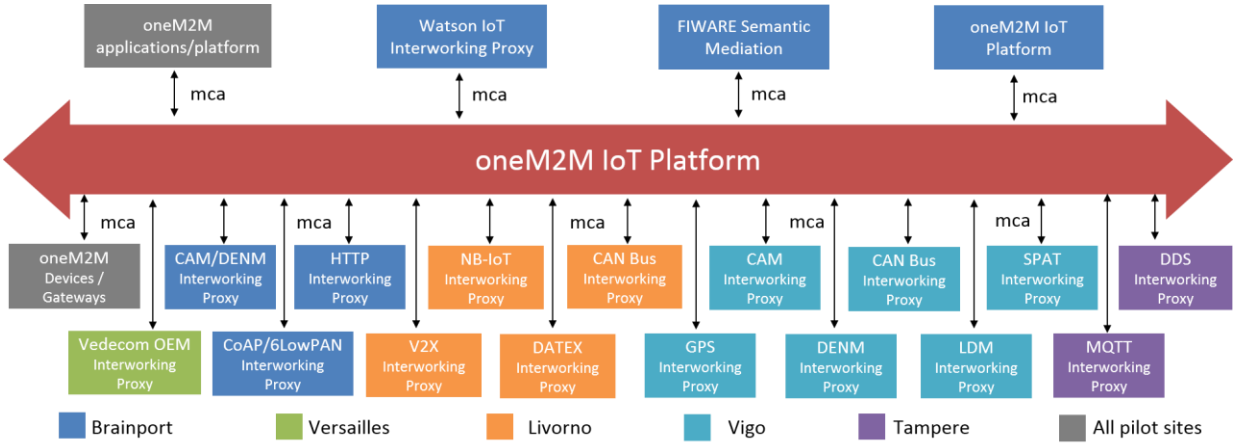


Figure 21: AUTOPILOT - required IPEs per pilot site overview [3]

The conceptual architecture is also used as the reference for discussion and collaboration with other LSPs.

The conceptual architecture will be utilizing oneM2M as standard based communication protocol and FIWARE enablers as service middleware. Among others, the AUTOPILOT project focuses on development and integration of IoT devices contributing to automated driving. These can be new devices or existing devices adapted to become IoT devices.

Mobile IoT objects (mobile robots and/or drones) and IoT infrastructure (sensor/actuators, connectivity and communication) will be developed and seamlessly integrated into the IoT ecosystem. In terms of development and integration,

AUTOPILOT is summarising all the interworking components required for IoT platform integration among the different pilot sites and their different application areas [3].

Mechanisms of connecting both oneM2M and Non-oneM2M devices/applications into the IoT platform are under development. A set of interworking proxies' entities (IPEs) are identified and shall be developed to integrate pilot site devices/applications to the oneM2M IoT Platform (see Figure 21).

4.1.3 Use Cases

The AUTOPILOT project is divided into five use cases, that will be demonstrated at six different pilot sites (see Table 1). Each pilot site includes between two and six use cases [1]:

- The urban driving use case requires automated driving vehicles to identify, predict and react in an array of complex situations. Fully automated vehicles will be tested for driving without any action from the driver. However, the driver will be able to override and get back to manual driving at any time;
- In the automated valet parking use case, the driver can leave and retrieve the car at some predefined locations. The operations of parking/retrieving and manoeuvring the car in the parking area, and possibly other additional services, such as fuelling, recharging or cleaning, will be managed by the parking management system;
- In the highway pilot use case, a cloud service merges the sensors' measurements from different IoT devices to locate and characterize road hazards. The goal is then to provide the following vehicles with meaningful warnings and adequate driving recommendations;
- The platooning use case demonstrates vehicular platoons consisting of a lead vehicle and one or more highly automated or driverless following vehicles which have automated steering and distance control to the vehicle ahead. The control is supported by V2V communication; and
- The real-time car sharing use case where the objective is to allow commercial and individual car sharing services to use automated driving vehicles. The service platform collects the end user needs and uses relevant data in the IoT platform to suggest car sharing possibilities.

Table 1: AUTOPILOT use cases versus pilot sites [3]

Use cases	Tampere (FI)	Versailles (FR)	Livorno (IT)	Brainport (NL)	Vigo (ES)	Daejeon (KR)
Automated valet parking	X			X	X	
Highway pilot			X	X		
Platooning		X		X		
Urban driving	X	X	X	X	X	X
Car sharing		X		X		

Below we give two architecture examples from the AUTOPILOT project. The first example is from the Italian pilot site which include the highway pilot and urban driving use cases as illustrated in Figure 22 [3]. The illustration gives an overview of the integrated uses cases using the oneM2M platform.

The IoT devices are integrated in the IoT oneM2M platform according to the oneM2M standard [3]. The IoT oneM2M platform is a federated model where several heterogeneous IoT platforms are interconnected.

A central IoT platform includes various modules: big data management and storage, real time and batch analytics, security and privacy, semantics, etc. Interoperability between the central IoT platform and the pilot site IoT platform is addressed in this platform.

The in-vehicle IoT platform is a component that provides a communication with the cloud IoT platform and the interfaces to other in-vehicle components.

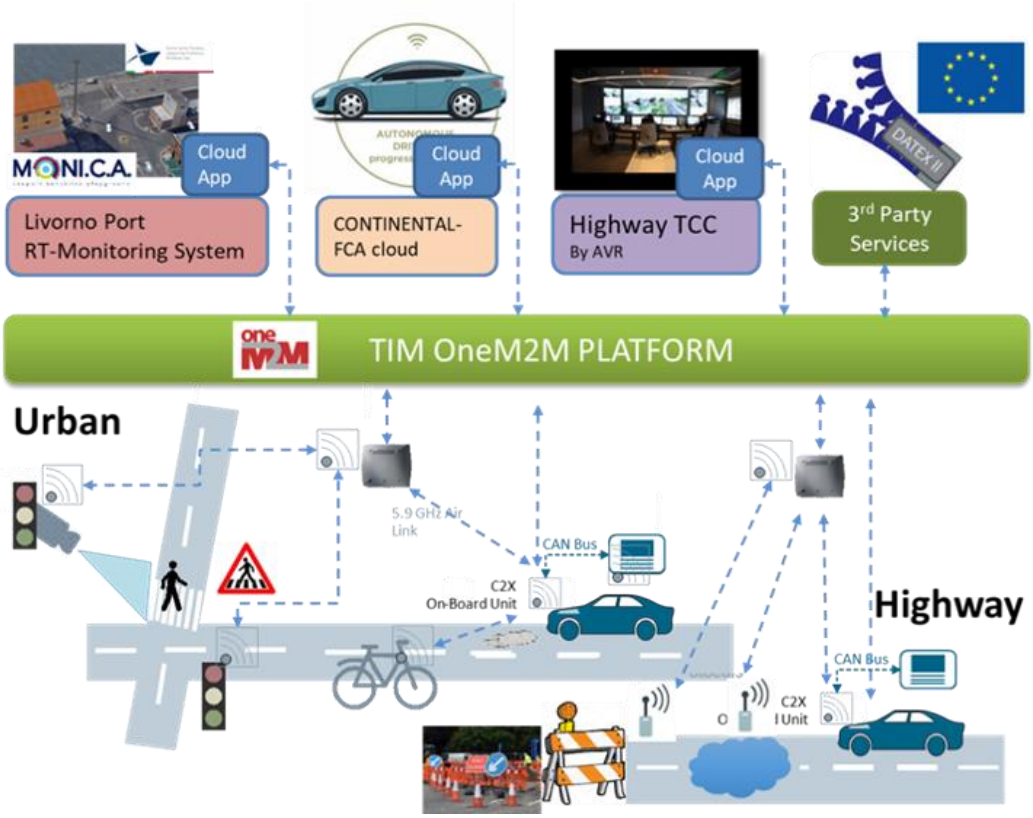


Figure 22: The Italian pilot site architecture (simplified) [3]

The second example is from the Spanish pilot site which include the automated valet parking and urban driving use cases. The pilot site has deployed a complex structure where many communication interfaces are involved, as illustrated in the communication architecture overview in Figure 23 [3]. However, both in-vehicle and central IoT platforms are oneM2M, the vehicle-to-central platform communication and the in-vehicle IoT communication are seamlessly forwarded, following both platforms the same messaging format.

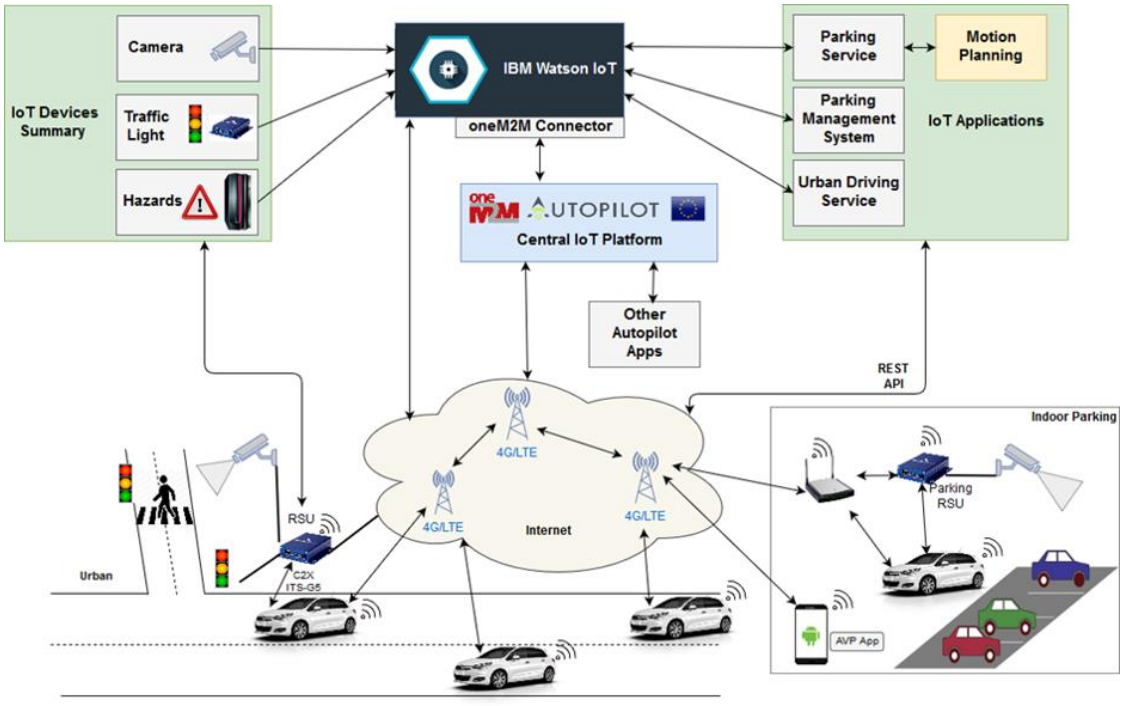


Figure 23: The Spanish pilot site communication architecture (simplified) [3]

The pilot site is composed by three main IoT platforms [3]: (i) The IoT platform is a federated model where several heterogeneous IoT platforms are interconnected. A central IoT platform includes various modules like big data management and storage, real time and batch analytics, security and privacy, semantics. Interoperability between the central IoT platform and the pilot site IoT platform is addressed in this platform. (ii) The in-vehicle IoT platform is a component that provides a communication with the cloud IoT platform and the interfaces to other in-vehicle components. (iii) The device's IoT platform could be one platform per device or group of devices. The devices can be new devices or existing devices adapted to become IoT devices able to be integrated into the IoT ecosystem.

In some cases, the pilot sites have chosen different solutions for the same purposes or functionalities. To show the big picture in terms of development and integration of IoT devices, the project is looking across the different solutions, to identify, compare and evaluate. All the interworking components required for IoT platform integration among the pilot sites will be summarized in the AUTOPILOT report on development and integration of loT device into loT ecosystem [3]. For instance, a set of interworking proxies' entities (IPEs) are identified and shall be developed to integrate pilot site devices/applications to the oneM2M IoT Platform. Some IPEs are duplicated between pilot sites, but in some cases, vendor-specific devices will communicate directly with specific IoT platforms using proprietary protocols.

4.1.4 Benchmarking

Several tasks in the AUTOPILOT project are dedicated benchmarking through collecting KPIs or metrics to address different perspectives or objectives of the project. For instance, Automated driving performance and safety KPIs (T4.2) and Quality of life KPIs (T4.4) relate to the topic of progress on benefits to the public; KPIs for scientific dissemination and project events organisation (T5.2); Business exploitation KPIs relating to the dependability, robustness, resilience, adaptability and sustainability of the piloted technology (T5.3) in order to validate business processes and models in relation to the AUTOPILOT's pilot sites and use cases; and KPIs for design, testing, validation and impact assessment for autonomous vehicles and IoT pilot impact measurement (T5.4).

The last one, Performance and KPIs for autonomous vehicles and IoT pilot impact measurement were prepared through the task "Cross-fertilization with loT and autonomous transport focus areas" [4]. This work focus on key performance indicators and common methodologies, to make recommendations about the main applicable KPIs for design, testing, validation, and impact assessment that shall be considered in the evaluation of the performance of the ecosystem.

In addition to objectives and high-level KPIs defined, this task goes deeper and analysis and provide an extensive number of KPIs for autonomous vehicles and IoT pilot impact measurement, categorize them into fields and map to the different use cases. 20 fields were identified, and descriptions, metrics, methods of collection/measurement, and project targets were prepared. Examples of fields addressed are: IoT tests, certifications, devices, modules, platforms, standards, monitoring, architectures, interoperability, scalability, maintenance, ecosystem awareness, engagement, and openness. Autonomous vehicles and IoT KPIs across application domains were also briefly addressed in this task.

Table 2: AUTOPILOT summary

Number (and name) of reference standards used for the architectural study	ITU-T SG13 Y.2060, ISO/IEC JTC1, oneM2M [23], FIWARE/ETSI
Does it provide a minimum reference architecture?	Yes, 4 layered architecture plus common reference architecture

Is the project considering the use of common standards?	Yes, NGSI, ETSI NGSI-LD, oneM2M
Are use cases/demonstrators asked to comply with the common reference architecture?	ITU-T SG13 Y.2060, ISO/IEC JTC1[22], oneM2M, FIWARE/ETSI
Are there evaluation KPIs related to use case compliance with architecture? Please name them	Open IoT architecture, Standard IoT architectures, Adherence with the AUTOPILOT in-vehicle IoT platform architecture

4.2 ACTIVAGE

The ACTIVAGE high-level architecture also known as conceptual architecture defines an interoperable IoT-enabled Ecosystem that support an Active and Healthy Lifestyle for the older citizens, it aims for providing better quality of life while ageing using technology and at the same time engage individuals into a culture of more healthy activities and enables and ecosystem of services around those activities and technology for healthy ageing.

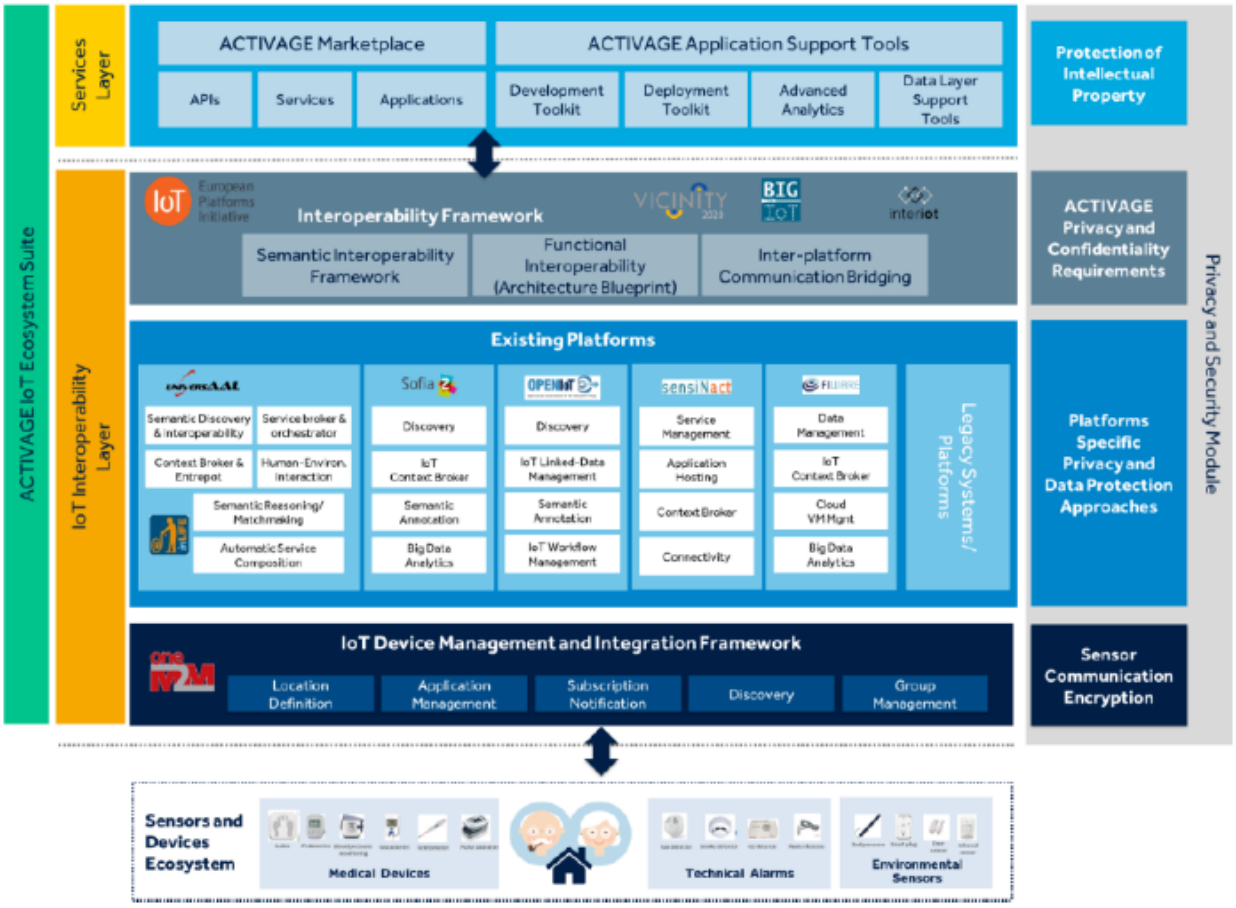


Figure 24: Conceptual architecture of the ACTIVAGE IoT ecosystem suite

The ACTIVAGE architecture separates the ACTIVAGE system into two distinct layers which are essentially based on diverse ecosystems, on one side IoT sensors and devices within a smart living environment that support indoor and outdoor activities/mobility and in the other side IoT services and applications using the data generated from the sensors and devices. By definition in the ACTIVAGE architecture Privacy and Security are by design and data protection plays a relevant role because of the personal data usage, making thus necessary to have a protection module along other management modules that are part of the full stack in ACTIVAGE ecosystem.

The two layers in the ACTIVAGE Architecture are comprised to support of a) universal interoperability framework for the integration of the widest possible spectrum of platforms in the area of Active and Health Living called AIOTES and b) the formation of the diverse application marketplace and a set of application tools for the support of creators in the development and deployment phases of new applications. Figure 24 summarizes the envisioned ACTIVAGE architecture that is also compliant with standardization projects such as IoT-A.

4.2.1 Reference architecture

The ACTIVAGE project defines as reference architecture the necessary set of IoT devices, systems, software components, set of services and APIs that can serve as common framework to build interoperable smart living solutions in the form of apps, software tools and services that can be deployed, extended and replicated at different geographical locations called deployment sites across Europe. In other words, ACTIVAGE support any additional IoT technology platform and services as far as they comply with the defined reference architecture, interoperability framework and standards.

The ACTIVAGE architecture is in compliance with the IoT-A reference model, and it has been designed to provide semantic interoperability, that enables and orchestrates the interconnection of heterogeneous IoT devices, open IoT platforms and smart living services within a common ecosystem of solutions. The architecture includes management components to support the deployment in geographical locations called deployment sites and provides a global API for the development of third party services and applications and a market place to locate them. Figure 25 present the ACTIVAGE simplified reference architecture where it can be seen that AIoTES API, located in the middle central part, is the enabler of API functions and management services that connects Applications with the technology. It is also important to highlight that the functions are interconnected with the interoperability layer to provide inter-connected data exchange.

The ACTIVAGE reference architecture is designed having in mind the need for efficiently and effectively integrate a wide spectrum of both open and commercial platforms and integrate IoT devices via the gateway approach, having as a starting point the platforms provided for the ACTIVAGE stakeholders and the IoT providers in the different Deployment Sites of the project.

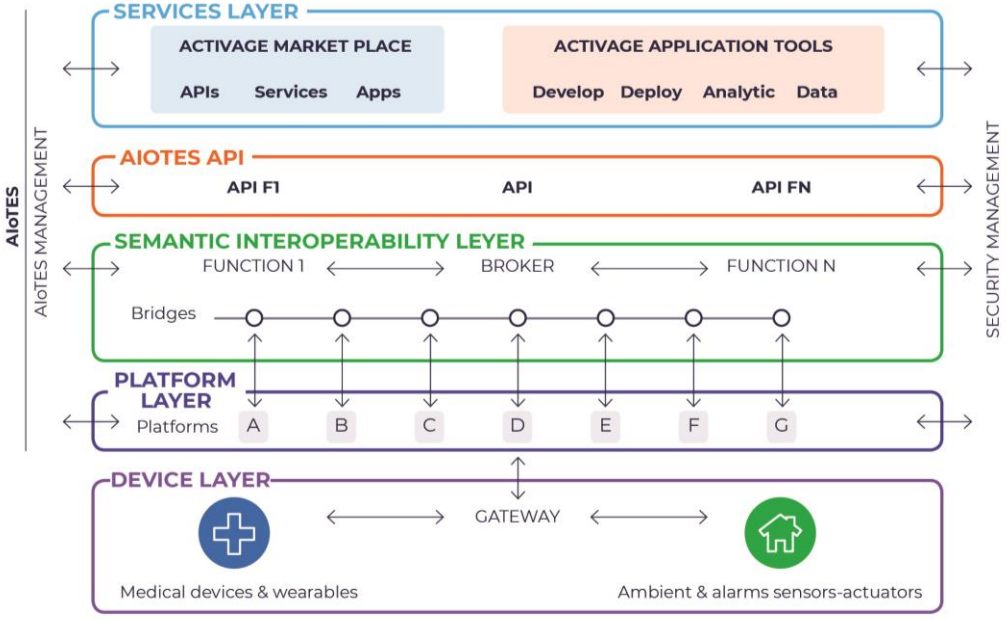


Figure 25: ACTIVAGE-AHA Reference Architecture Simplified.

The ACTIVAGE IoT Interoperability Layer

The IoT Interoperability Layer is aiming to efficiently and effectively integrate a wide spectrum of open and commercial platforms and IoT devices, having as a starting point the platforms provided for the ACTIVAGE and the IoT devices used in the Deployment Sites of the project.

The Interoperability layer has been proposed as a solution to overcome the lack of interoperability among the existing IoT platforms. Each platform has different standards and data formats. For these reasons, interoperability, data sharing and communication among platforms is considered one of the most arduous challenges in IoT as it is represented in Figure 26.

This component provides an abstraction layer at middleware level, which will be connected to all the IoT platforms deployed in the ACTIVAGE Project. This layer, that is a middle element among the IoT platforms and ACTIVAGE, will allow the communication and information sharing among those platforms and ACTIVAGE (e.g. collection of data from AHA sensors).

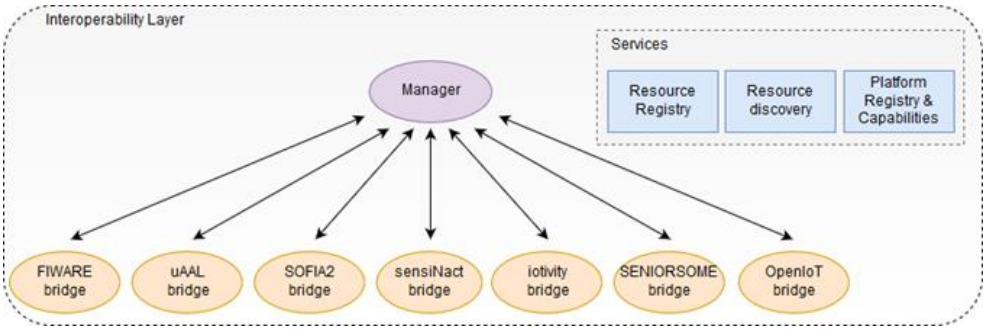


Figure 26: Interoperability Layer and Platform Bridges

The ACTIVAGE IoT Interoperability layer can be further separated into two frameworks that will create standardized interfaces a) The IoT Device Management and Integration Framework for the sharing of data with sensors and devices and b) The Interoperability Framework which offer data interoperability services and takes care of security and privacy by design in order to meet the different regulations.

a) The IoT Device Management and Integration Framework will be responsible for the conformity of the ACTIVAGE ecosystem with widely accepted standards for device communications within a network of IoT. The results and developed standards of OneM2M are the foundation upon which the integration and management of devices will be done, offering advantages such as sensor discovery and location definition as well as application management and subscription notifications.

b) The Interoperability Framework aims for providing the required components for the operation and communication of all the integrated platforms. More specifically three dimensions of interoperability will be covered with respective services, namely: 1) Semantic Interoperability related to the data shared within the system, 2) Functional interoperability as it relates to the required functionalities that all the platform of the ACTIVAGE ecosystem should implement and finally, 3) Inter-platform communication bridging. Previous experience of the consortium in related projects such Sofia2, UniversAAL and Vicinity will serve as the basis for these interoperability components. Furthermore, results of IoT standardisation efforts such as BigIoT and InterIoT will be used for the development of all related components.

The ACTIVAGE Services Layer

The top layer of the ACTIVAGE system include several functionalities to support efficient integration and effective deployment of new services to the envisioned ecosystem. The Services Layer: the top layer of the ACTIVAGE system aims for including A set of functionalities to

support efficient integration and effective deployment of new data services to the envisioned ecosystem of citizens for Active and Healthy Ageing (AHA). The ACTIVAGE services layer can be understood as a set of APIs that ca be further classified into two set of different interfaces. 1) the ones dedicated to support services in the context of an ACTIVAGE Marketplace and 2) the ones dedicated to support applications, development and analytics services.

1) The ACTIVAGE Marketplace combine systems and data interoperability characteristics and service-oriented development in a common AIoTES framework where developers could publish their products and users can combine applications and individual services in a cloud-based environment. Within this framework users are able to discover, and match pilot applications and service offers with their needs, whereas business entities will be able to publish their offers and collect review and rate for their products. In this direction ACTIVAGE marketplace is able to support the international outreach of the project, underline the openness of the ACTIVAGE framework and support new efforts aiming at Active and Healthy Living in terms of fast and effective dissemination.

2) The ACTIVAGE Application support tools are designed for creators in the development and the deployment of processes towards new applications using specialized approaches. More specifically the development toolkit of the project cover registration, resolving and discovery of services through a Web-Based Software Development Kit allowing developers to rapidly configure new products. Furthermore, advanced analytics tools as well as a data integration engine will allow the understanding of spatiotemporal patterns that can help developers maintain and optimize their products. Finally, and in order to support creators through the deployment processes, a toolkit of testing, validation and upgrading functionalities will be implemented and will be fundamentally used in the pilot use cases as they are scheduled by the project.

The ACTIVAGE Security and Privacy protection module

This ACTIVAGE Security and Privacy protection module is a crucial element with a set of different components of the ACTIVAGE IoT Ecosystem Suite that spans across all the above layers and suites and its purpose is to guarantee both the protection of sensitive information of users and will also comply with ethical and legal requirements for privacy and confidentiality. Furthermore, the security module is the responsible for the protection of the intellectual property of application developers.

The ACTIVAGE Security and Privacy protection module will be supplied with intrinsic security, trustworthiness and privacy by design methods by securing trustworthy cooperation and scalability of requirements, through the formulation of a secure and trusted IoT cloud environment based on innovative utility-based schemes.

The project emphasizes the secure cooperation and scalability of requirements, through the formulation of a secure and trusted IoT cloud environment based on innovative utility-based schemes.

The security and privacy protection are crucial components of the AIoTES framework since they span across all the above layers and components and guarantee both the protection of sensitive information of users and also to comply with ethical and legal requirements for privacy and confidentiality. Thereby, the security module is responsible for the protection of the data against unauthorized access.

As part of the associated activities on the Security and Privacy protection module; the creation and maintenance of the Ethics and Privacy Protection Manual will guide the principles and the main procedures regarding privacy, data protection, legal issues and ethical challenges. All legal and ethical issues regarding individual privacy, trust and security will also be taken into account.

4.2.2 Architectural focus

The ACTIVAGE project Architecture focuses on delivering an IoT Ecosystem Suite (AIoTES), a set of Techniques, Tools and Methodologies for interoperability at different layers between heterogeneous IoT Platforms and an Open Framework for providing Semantic Interoperability of IoT Platforms for AHA, addressing trustworthiness, privacy, data protection and security.

AIoTES and Interoperability are means for user-demand driven interoperable IoT-enabled Active & Healthy Ageing solutions and the intention is to motivate that multiple AHA applications can be deployed on top of the AIoTES in every DS, enhancing and scaling up existing services, for the promotion of independent living, the mitigation of frailty, and preservation of quality of life and autonomy. ACTIVAGE will also focus on assessing the socio-economic impact, the benefits of IoT-based smart living environments in the quality of life and autonomy, and in the sustainability of the health and social care systems, demonstrating the seamless capacity of integration and interoperability of the IoT ecosystem, and validating new business, financial and organizational models for care delivery, ensuring the sustainability after the project end, and disseminating these results to a worldwide audience.

ACTIVAGE architecture, along with all its necessary functional components and its multiple interoperability interfaces is shown in Figure 27, from this architecture the AIoTES framework has been defined. This framework consists of four main blocks, namely, the AIoTES Management block, the Privacy block, the Security block and the IoT Semantic Interoperability Layer (SIL) along with the AIoTES API.

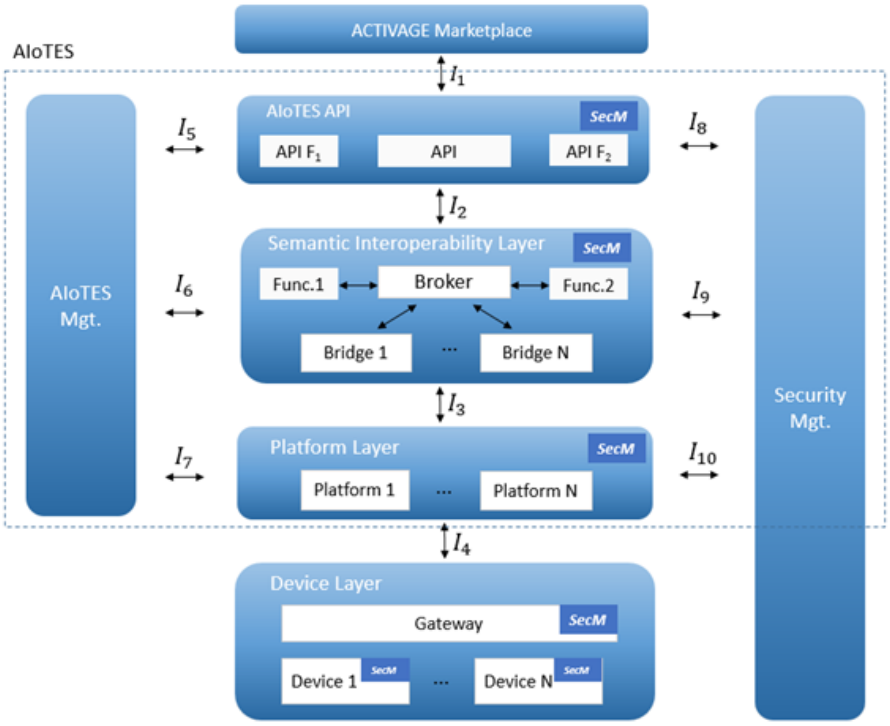


Figure 27: ACTIVAGE-AIoTES Functional Components and Interoperable Interfaces

Starting at the top of the AIoTES framework, the AIoTES API is a common API which offers a homogeneous access to the ACTIVAGE features regardless the configuration or particularities of the scenario. It enables interaction between the applications of the Marketplace and the different components to the AIoTES Framework through the APIs provided by the components.

The most remarkable benefit of the AIoTES API is that it will make it possible to reuse and exchange heterogeneous services from the different IoT platforms and allow application developers to produce new added value services from existing IoT services. Thereby, the use of this API will allow third parties to develop new application and services compatible with the ACTIVAGE paradigm and contribute to the creation of the development ecosystem.

The AIoTES Management can be defined as the set of infrastructure and tools through which users can obtain information related to DSs (e.g. platforms and sensors). It aims at providing mechanisms, tools and helper contents to make proper use of the IoT Semantic Interoperability layer in addition to help with the fully integration of the AIoTES Framework.

Interoperability through the IoT platforms presented in ACTIVAGE project is carried out by means of the IoT Semantic Interoperability Layer (SIL) which is an abstraction layer that allows the communication between an application of the marketplace and ACTIVAGE platforms. This layer involves processing, storage and management utilities together with bridges that connect with several other utilities located in the platforms that are being interoperated.

4.2.3 Use Cases

ACTIVAGE project also considered the project for "Ageing Well with IoT" is considered as the mean for extending healthy living years of older adults living independently and autonomously in their preferred environments by the adoption of immersed/embedded IoT devices and solutions.

The use cases in ACTIVAGE are user-driven needs. ACTIVAGE focusses on "domains of needs" for the support of the older population and in order to create a demand-driven experience on the basis of the reference Ageing Well initiatives around the world. The Figure 28 shows the user-driven needs in the Concentric circles and numbered from 1-9 the uses cases associated to them.

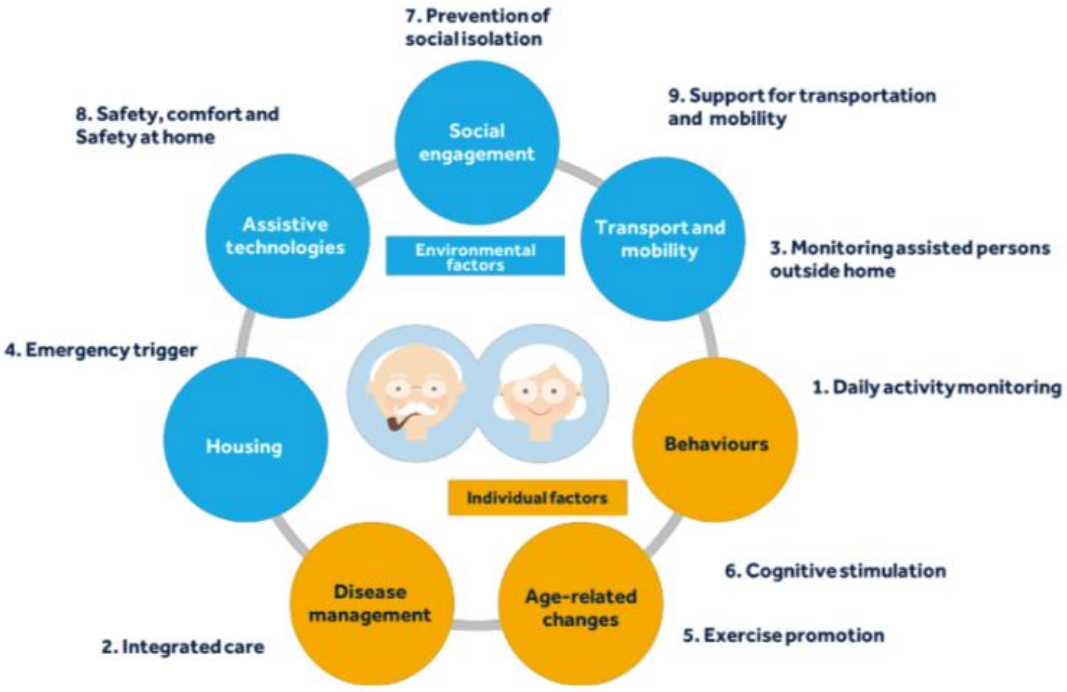


Figure 28: ACTIVAGE Domain Areas as User-Driven Needs & Use Cases

The following Table summarizes the challenges that at the same time are specific use cases by domain that ACTIVAGE aims to address through IoT-based services and solutions supported by AIoTES framework. The Table 3 Summarize the ACTIVAGE use cases and Domain Areas.

Table 3: ACTIVAGE use cases and domain areas

Domains of needs of older people		ACTIVAGE addressable challenges (CHx) by means of IoT-based technologies
Individual factors	Behaviours	**ACTIVAGE Use Cases: 1. Daily Activity Monitoring.** Early prevention of mental, behavioural, and health-related decline in older people living by introducing seamless behavioural assessment and monitoring devices and intelligent algorithms both at home and in smart city environment.
	Age-related changes	**ACTIVAGE Use Cases: 5. Exercise Promotion and 6. Cognitive Stimulation.** To give support to older people in their daily lives as a consequence of the ageing process by means of sensors, actuators and other specific (i.e. exercising) connected equipment in order to mitigate the effect of ageing and fighting against physical decline in smart living environment.
	Disease management	**ACTIVAGE Use Cases: 2. Integrated Care.** To support patients and healthcare and social care professionals in integrating social care and support with health care for a more effective management of the elderly person, in order to go towards the demonstration of benefits for the person and savings to the healthcare system.
Public/Group Factors	Housing	**ACTIVAGE Use Cases: 4. Emergency Trigger.** To support the elderly person in mitigating the risks of a house that is not evolving with them to make their life more comfortable thanks to the deployment of devices, actuators and adapted user interfaces to seamlessly control the house conditions.
	Assistive technologies	**ACTIVAGE Use Cases: 8. Safety Conform and Security at Home**. To complement the functional capacity of the end users with tools that help them to seamlessly operate daily living activities both at home and outside, by means of sensitized wearable elements, and ambient infrastructure that is networked to offer help and support the needs both at home and outside home.
	Transport, mobility and Leisure	**ACTIVAGE Use Cases: 3. Monitoring Assisted Persons Outside Home and 9. Support for Transportation and mobility.** To empower elders to move across different areas and visit new places, through advanced mobility and leisure services within a smart IOT ecosystem that will minimize the danger of reduced care options or the requirement of a transitioning period that may affect negatively their active leaving and health.
	Social engagement	**ACTIVAGE Use Cases: 7. Prevention of Social Isolation**. To keep older people actively involved so they still (believe that) contribute back to the society and participate from its events; and to keep away the negative effects of decline and depression that are the result of lack of mental and physical activity.

4.2.4 Benchmarking

ACTIVAGE is a demand-driven project and as such there is active participation from the different stakeholders in an ecosystem, i.e. citizens, services providers, systems and technology providers.

producers of data processors and consumers of information. etc. this involvement implies to have identified figures that are also the metrics to measure success. The following Table 4 summarize the ACTIVAGE project measurable results.

Table 4: ACTIVAGE measurable results

Metric	Main Result	Verification (success criteria)	KPIs
Diversity	Use Cases, Services and Business Cases built, demonstrated, expanded and grown across all the 9 Deployment Sites of the 7 European Countries.	Project Successfully Executed and Evaluated / at least 35 Value-Added services demonstrated and evaluated by end- users during pilot realisation / Number of stakeholders approached	≥9 Deployment Sites. ≥35 Value-Added services
Inter-operability and Usability	Semantic interoperability layer allowing integration and interoperability of heterogeneous platforms; Framework and API allowing the connection of new services and interact transparently with IoT platforms.	Number of platforms integrated to ACTIVAGE Number of connected infrastructures (At least 5 IoT systems integrated to ACTIVAGE)	≥5 IoT systems integrated
Scalability	Implementation, Demonstration and Replication of Use Cases; Execution of Business Cases from internal and external stakeholders.	Percentage of business and legal requirements covered by the ACTIVAGE	TBD in Phase 3
Adoption and Ecosystem Enlargement	ACTIVAGE GLOCAL evaluation framework: Pilot executed across all the Deployment Sites; Local Key Performance Indicators Collected; Global Key Performance Indicators Collected	Evaluation Open Data Base developed, validated by Internal and External Deployment Sites. White Book delivered.	TBD in Phase 3
Security and Innovation	Co-creation framework assessing needs, preferences and perceptions of ACTIVAGE users on acceptability, trust, confidentiality, privacy, data protection and safety.	Users Involved in the Design and Evaluation Phases. Indicators of: Usability of the Solution (ISO 9241), User Experience, Willingness to Buy, above threshold.	≥9000 users across all the deployment sites
Engagement	Communication and dissemination program, networking activities connected worldwide.	Compliance with the expected value of the indicators (web activity, Google alerts, social network activity, targeted activities and media presence)	TBD in Phase 2
Socio-Economic Impact	Market growth and sustainability based on ACTIVAGE replication pilots and new investment in SMEs, start-ups and IoT providers.	Number of 3rd party services integrated to ACTIVAGE (At least 3 IoT infrastructures integrated beyond the current consortium / at least 3 new services integrated in ACTIVAGE	≥3 IoT external infrastructures ≥3 new services integrated

Summary of KPIs set for evaluation

Table 5: ACTIVAGE summary

Number (and name) of reference standards used for the architectural study	IoT Platforms IoT Devices IoT technologies
Does it provide a minimum reference architecture? Yes, it brings together different domains and provide reference design and implementations and at the same time expand the ecosystem of IoT in Healthcare and Viceerse, Healthcare domain using more IoT technology.	ACTIVAGE High Level or Conceptual Architecture AHA Reference Architecture AIoTES Functional Architecture AIoTES Simplified Architecture
Is the project considering the use of common standards? Yes, several form different organisations, some of them implicit in the technology other ones developed in the context of software and/or technologies.	REST, RESTful OAUTH2, HTPPS, TLS, DTLS, HL7 ITU-T, oneM2M, IETF, TMF
Are use cases/demonstrators asked to comply with the common reference architecture?	Yes
Are there evaluation KPIs related to use case compliance with architecture? Please name them	Yes, See Table 3 above

4.3 IoF2020

IoF2020 follows a multi-stakeholder approach, whereby trials and use-cases form the core of the project and are the basis for knowledge and applications development.

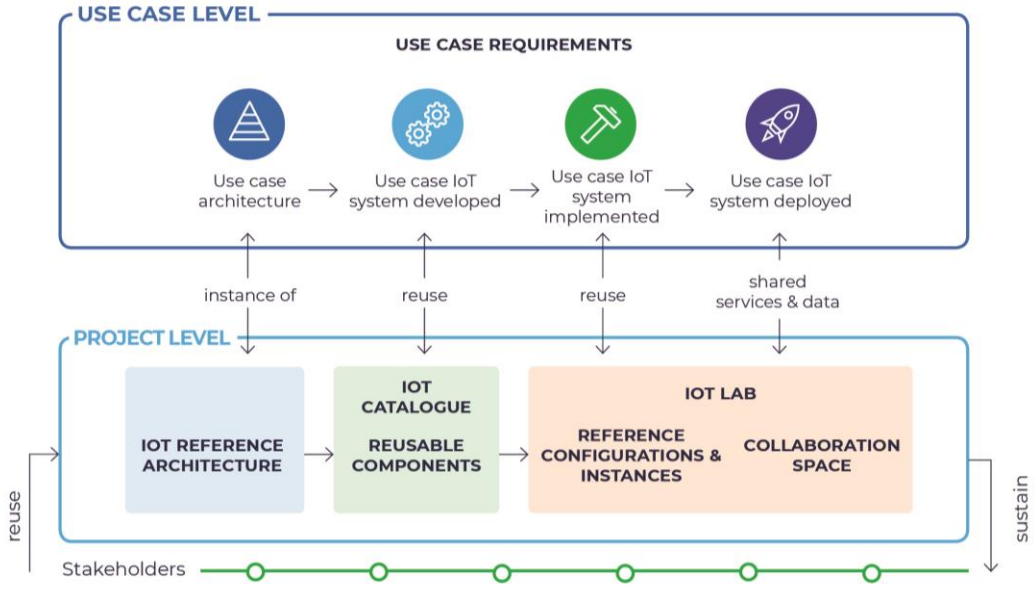


Figure 29: IOF2020 architecture simplified.

Stakeholders from research and business organisations work in close collaboration to quickly develop minimum viable products and create synergies through technical integration, governance and business modelling, and ecosystem development. IoF2020 focuses on interoperability and aims to provide a catalogue of re-usable system components, which can be integrated in the IoT systems of multiple use-cases of the project.

The following content is extracted from IOF2020 deliverable D3.1 GUIDELINES FOR USE CASE ANALYSIS & DESIGN, which was published back in September 2017 and contains the approach and reference architecture to be applied for the analysis, design and conception of IoT based solutions

4.3.1 Reference architecture

IoF2020 has selected ITU-T Y.2060 IoT Reference Model as baseline for all project activities (see section 3.1.6).

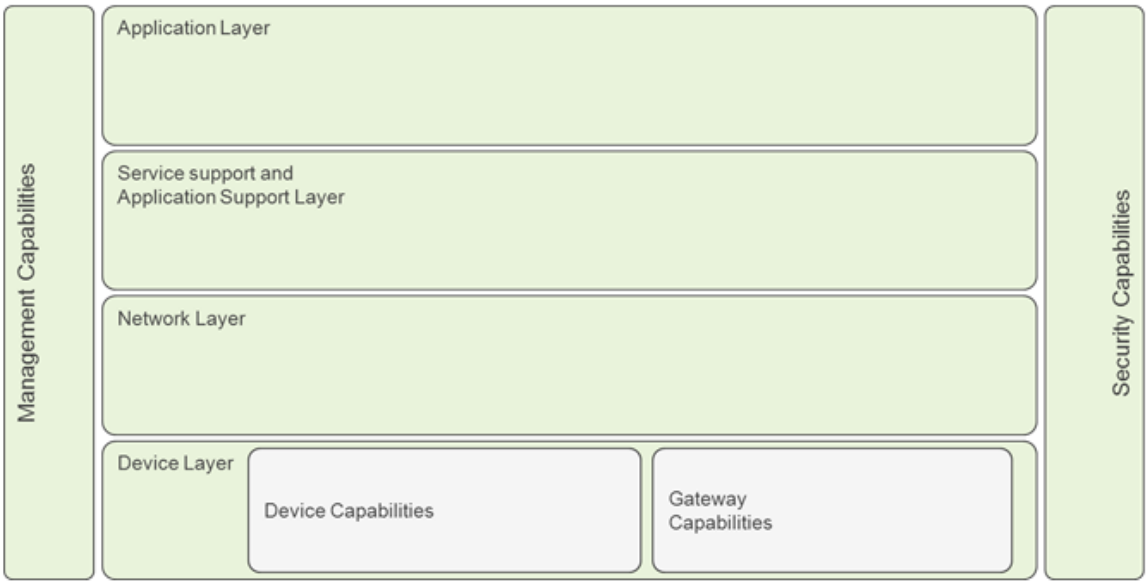


Figure 30: The ITU-Y Y.2060 Reference Model.

The selection has been made through and alignment effort, studying different IoT trends and standardization efforts and using AIOTI WG03 as reference. The high-level architecture adopted can be seen in Figure 30. Once the reference architecture is selected, IoF2020 started the work of matching the project components with respect to the reference IoT functionalities. This way, component features were analysed to select the most appropriate layer so that it could be included. As a result, for each use case present in IoF2020, an instance of Figure 30 is provided with the component correspondence and a short description.

4.3.2 Architectural focus

Being IoF2020 a cross-sectorial project using horizontal IoT approaches (in the form of IoT applications), the focus is put on selecting key architectural aspects, generic enough, to cover the needs of all application and uses foreseen.

Therefore, IoF2020 started from Use Cases, analysing their need, and then selected the most suitable architecture. This selection is backed up by the analysis of scientific and technical guidelines, standards and recommendations.

An ultimate and universally accepted methodology and architecture for complex and cross-cutting IoT use cases is something that has not yet emerged. Nevertheless, successful best practice recommendations can be found, being ISO/IEC/IEEE 42010 international standard for “Systems and software engineering - Architecture description” the selection made.

Using this reference as baseline, and guided also by AIOTI WG03, IoF2020 has adopted the practices and suggestions included in the recommendation, taking the decision to map each use case towards a common High-Level Architecture (HLA) model.

4.3.3 Use Cases

IoF2020 defines a set of multi-actor trials that reflect the diversity of the food and farming domain, including different actors and different supply chain roles, like logistics and consumption. The trials are composed of use cases, selected in interaction with the agri-food community, which address the most relevant challenges for the specific subsector concerned. The project has 19 Use Cases divided in 5 trials spread across Europe, as depicted in Figure 31.

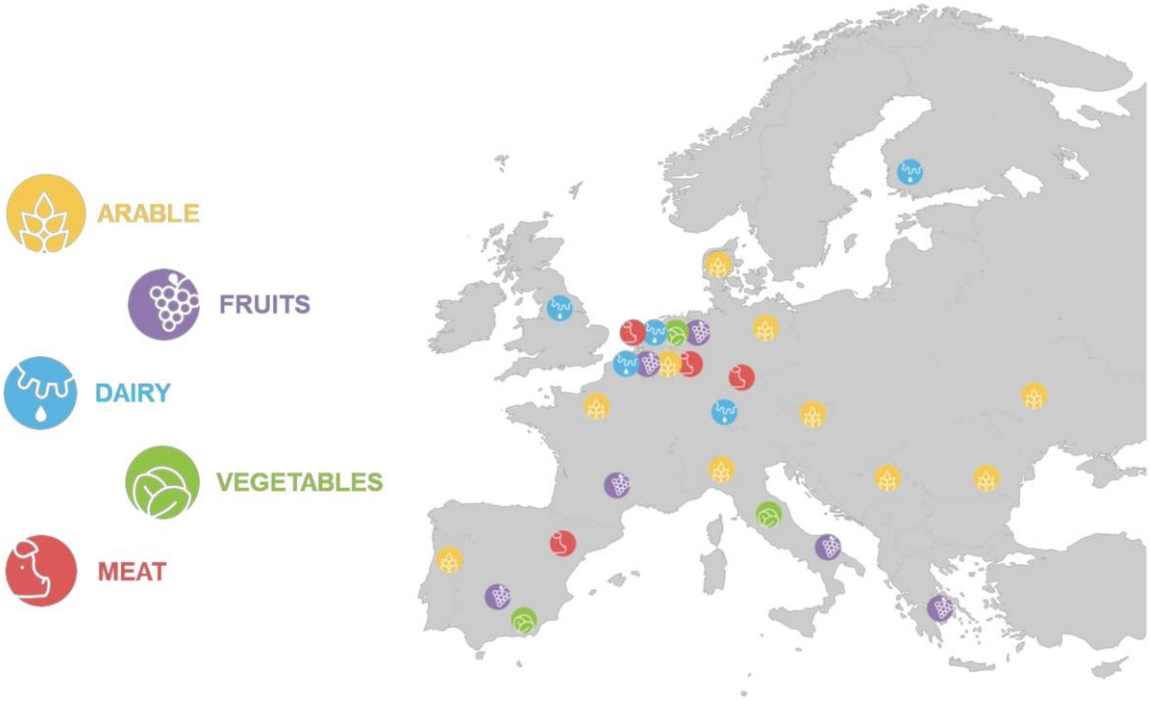


Figure 31: IoF2020 Trials and Use Cases

Figure 32 depicts the overall process followed in IoF2020 to elicit a common IoT architectural description within the IoF2020 project.

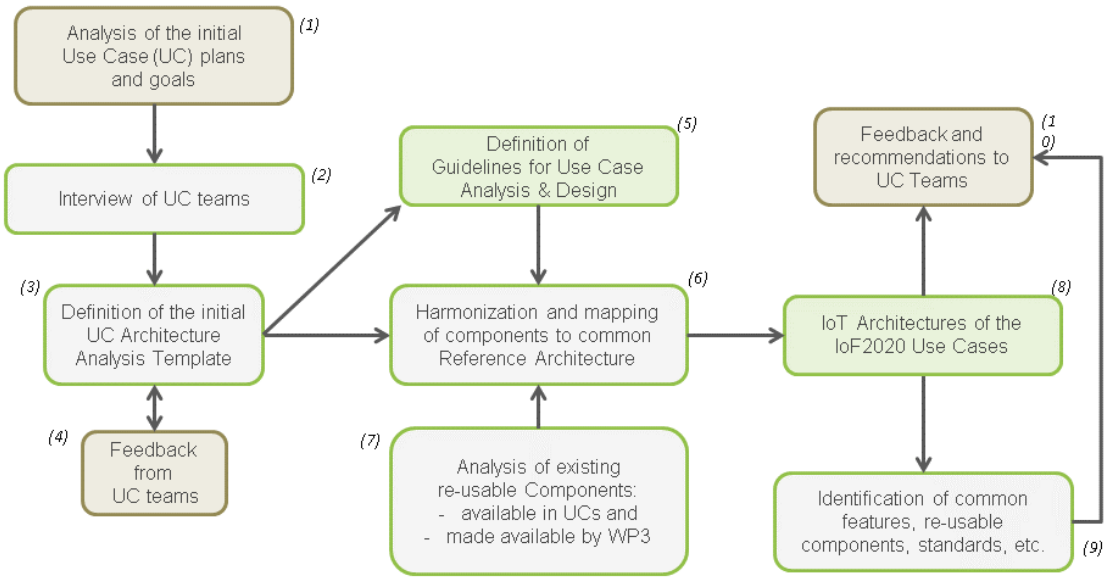


Figure 32: Overall analysis and design methodology.

IoF2020 is a use case-oriented project, so the work started and that point with a high-level description of use cases; as pointes in the step 1 of the Figure 32.

A technical team and a group of analysts elaborated a set of open questions related to these high-level descriptions, with the objective of getting architectures from use cases.

The outcome of these interviews was a template named "Use Case Analysis template". This template was mainly filled by UC teams with no strong background in IoT and/or system engineering but supported by a technical group.

The template was specially structured to allow them to provide all the necessary technical aspects in a simple format, ensuring enough information is gathered.

This information was later harmonized and mapped to a functional Reference Architecture. Also, this document detected possible gaps (where the use case teams required assistance to find suitable IoT solutions) that needs to be developed.

This methodology and the subsequent iterations between use case teams (a dedicated team member by each use case, named UC analyst) and technical groups allowed to produce a final methodology for IoT architecture analysis and implementation. T

he UC analyst was responsible to develop different views of the IoT architecture, to report the status of the analysis and identify synergies across the use cases, interact with all the teams and to review the security analysis.

The goal is to identify re-usable components that can then be re-used by other Use Cases. However, the main challenge was not to identify the re-usable components within each of the tiers, where the goals are at some extent similar, the biggest challenge was to identify re-usable components across different tiers, where the whole architecture was very different and the similarities with use cases in different tiers were next to none.

This methodology is considered as an "IoT pull" analysis with other activities in parallel to enable possible "IoT push" actions, where existing IoT solutions could be potentially available for use cases.

4.3.4 Benchmarking

Table 6: IoF2020 summary

Number (and name) of reference standards used for the architectural study	1 (ITU-T SG13 Y.2060)
Does it provide a minimum reference architecture?	Yes
Is the project considering the use of common standards?	No
Are use cases/demonstrators asked to comply with the common reference architecture?	Yes
Are there evaluation KPIs related to use case compliance with architecture? Please name them	Not yet

4.4 MONICA

The MONICA IoT platform consists of control systems which monitor the data collected and which can perform automated actions. Components analyse data and detect critical incidents, supporting operators in assessing the situation and making decisions. To ensure data security and trust, the solutions are built on Privacy by Design principles.

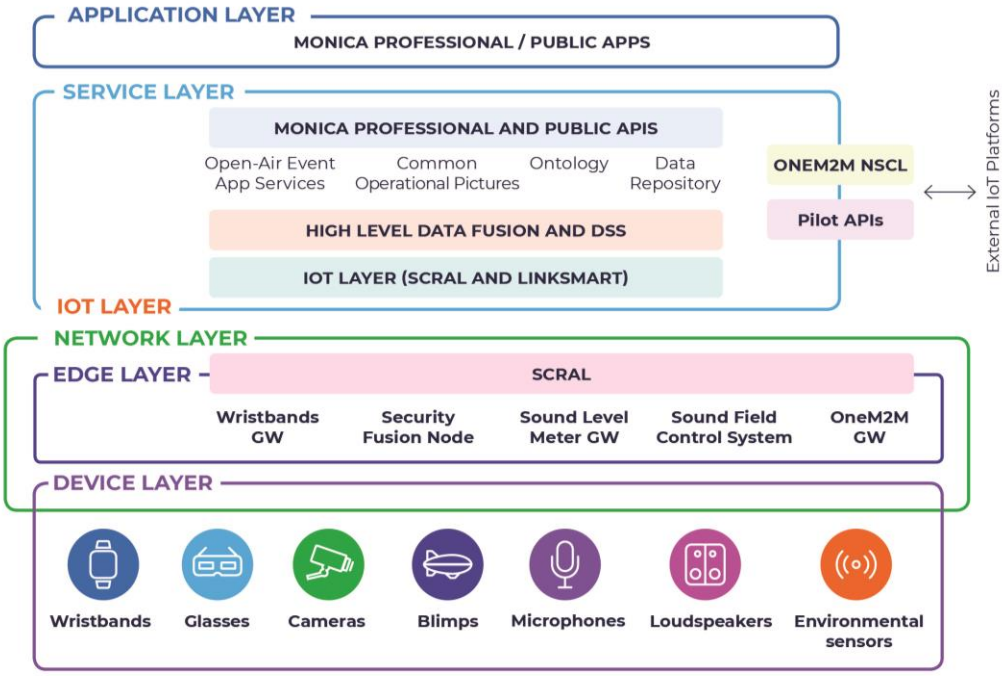


Figure 33: MONICA architecture simplified.

Based on open standards and architectures, the platform can be incorporated with existing smart city systems, replicated to fit other settings or used to develop new smart city applications.

4.4.1 Reference architecture

The MONICA architecture has been defined following the HLA developed by the Working Group 3 (WG3) of the AIOTI. The WG3 last updated the HLA in June 2017 (AIOTI HLA, 2017), which at the same time is described using the ISO/IEC/IEEE 42010 standard (ISO/IEC/IEEE 42010, 2011). Related documents to the HLA defined by the WG3 are available at [53].

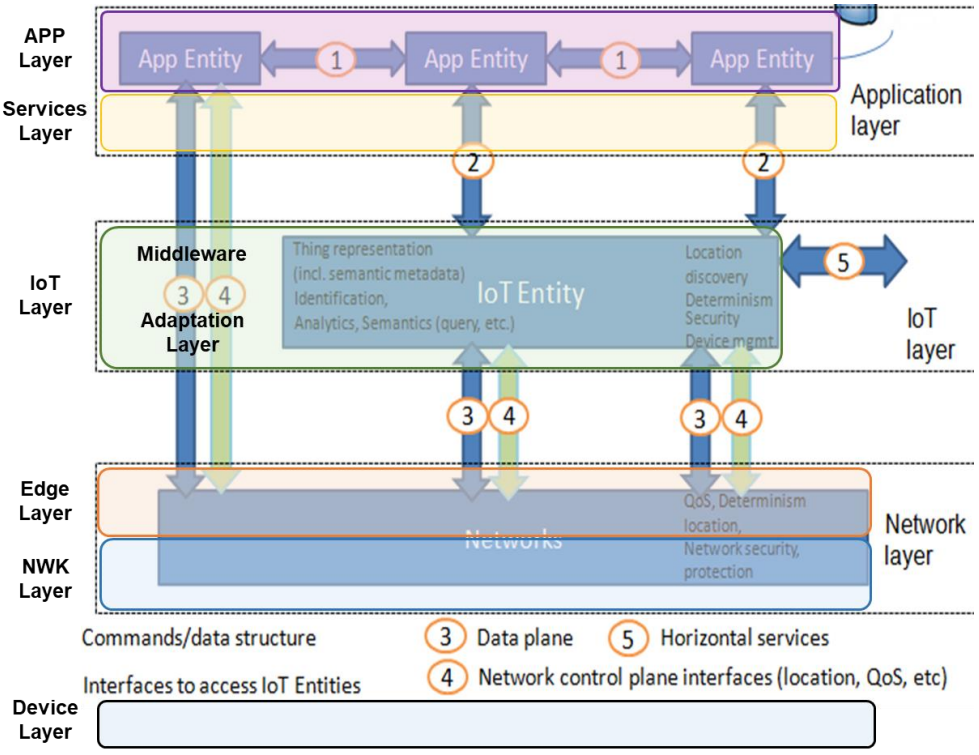


Figure 34: MONICA HLA Mapping with AIOTI HLA.

The functional model of AIOTI is composed of three main layers:

- The **Application layer** containing the communications and interface methods used in process-to-process communications.
- The **IoT layer** that groups IoT specific functions, such as data storage and sharing, and exposes those to the application layer via interfaces commonly referred to as Application Programming Interfaces (APIs). The IoT Layer makes use of the Network layer's services.
- The **Network layer**, which services can be grouped into data plane services, providing short and long-range connectivity and data forwarding between entities, and control plane services such as location, device triggering, QoS or determinism.

Figure 34 depicts a mapping of the current MONICA Architecture with the reference HLA defined by the WG3 of the AIOTI [11].

As it can be seen, the MONICA Network Layer and Edge Layer are seamlessly mapped with the Network layer of the reference HLA, enabling interconnections between heterogeneous networks and providing device connectivity to the Internet via different network technologies.

The MONICA IoT Layer, which includes the Adaptation Layer and the Middleware Layer, is mapped into IoT entities, since they provide IoT functions to App Entities or other IoT Entities via LinkSmart.

This layer uses the underlying Networks' interfaces to send and receive data from the physical devices (Device Layer of the MONICA architecture) that are not aware of the IoT world. The AIOTI HLA scheme does not explicitly include a Device Layer but this can be considered as part of its Network Layer while MONICA HLA explicitly represents it with a dedicated layer to highlight its importance within the project.

Finally, the MONICA's APP Layer and API's layer are mapped into the AIOTI HLA APP Entities since they implement - and enable - application logics.

4.4.2 Architectural focus

The current version of the MONICA architecture is depicted in Figure 35.

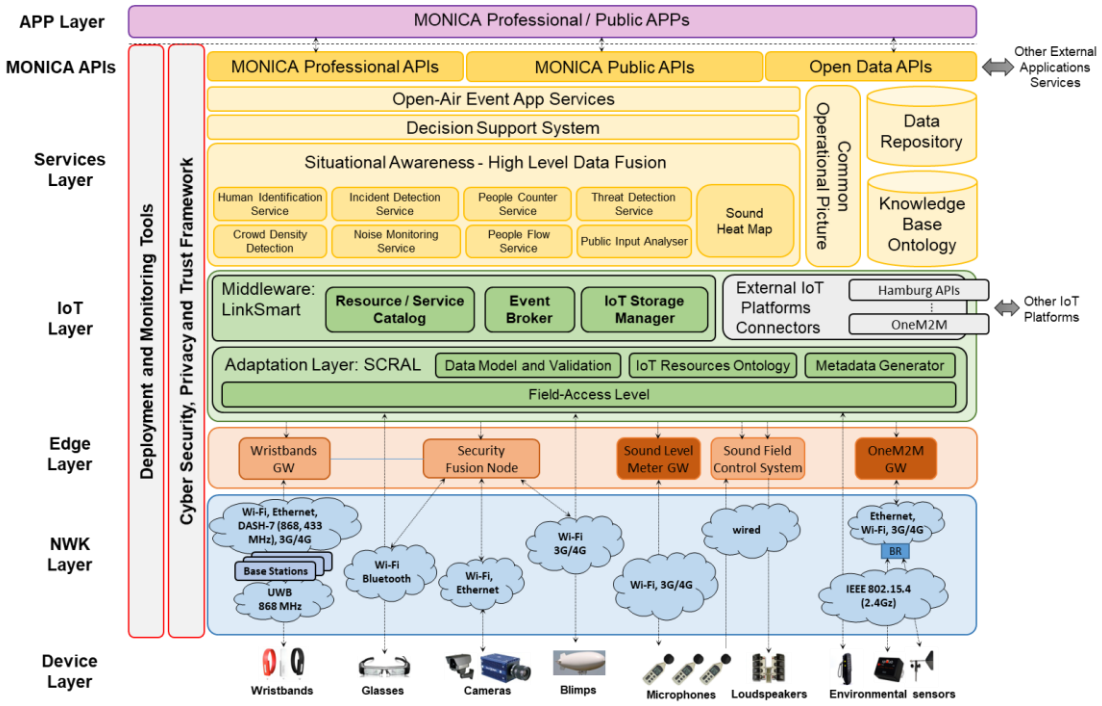


Figure 35: Functional View of the MONICA Architecture.

As it can be observed, the architecture comprises the following subsystems, also called layers:

- The ***Device Layer*** includes all IoT wearables (*e.g.*, wristbands and glasses) and IoT sensors, which can be fixed (*e.g.*, sound level meters, loudspeakers, cameras, environmental sensors) or mobile (*e.g.*, wireless sound level meters, cameras installed in a Blimp).
- The ***Network Layer*** that allows the effective communication between the heterogeneous IoT wearables, IoT devices and the IoT platform modules. This layer is responsible of forwarding data coming from the IoT wearables and IoT sensors as well as of responding to service requests coming from upper layers;
- The ***Edge Layer*** includes a set of processing modules (*e.g.*, the Wearables GW running localization algorithms, Processing Units executing video-based algorithms, the Sound Field Control System (SFCS) for managing the sound quality and noise reduction) that process real-time data directly from the *Device Layer*. To this purpose, these modules need to be deployed locally in the pilot site to avoid the latency introduced by the upper layers of the platform. Moreover, these modules require an efficient and scalable Network Infrastructure.
- The ***IoT Layer*** is composed of the following three subcomponents:
    - The *Adaptation Layer*, here represented by the SCRAL, providing technology independent management of physical resources and uniform mapping of data into standard representations that can be easily handled by the upper platform modules;
    - The *Middleware*, here represented by the LinkSmart, which offers storage and directory services for resources registered in the IoT platform;
    - The *External IoT Platform Connectors*, handling the communication with external IoT platforms and the integration of data coming from outside (*e.g.* from the Hamburg Smart City platform). In addition, MONICA integrates the OneM2M interfaces (Mca and Mcc/Mcc') allowing the platform to expose the IoT data according to the OneM2M standard;
- The ***Services Layer***, where the intelligence of the platform is implemented, and specific processing modules are integrated to provide technical solutions compliant with the application requirements. The services modules are combined together with knowledge base components and decision support tools whose aim is to propose a set of intervention strategies to assist human operators in gathering context-sensitive information and decision making;
- The ***MONICA APIs Layer***, which provides service access points for MONICA application developers and external application developers that want to access MONICA functionalities and information streaming from the platform;
- The **Cyber *Security and Privacy Framework***, enabling trust-based communication, policy management and technical support across all levels of the platform. More specifically, this framework ensures secure data flows and storage, protected information exchange and trusted federation mechanism to facilitate private information sharing;
- The ***Deployment and Monitoring Tools.*** These tools belong to a transversal framework able to easy the platform deployment (*e.g.* modules belonging to the *Device* and *Network* layers) and used for checking the operational status of the devices, networks and overall system. Moreover, these tools are also used for measuring performance metrics and tracing pilot events.

4.4.3 Use Cases

MONICA project adopts an iterative approach to define use cases and requirements to implement use cases across all pilots.

The following picture shows MONICA pilots where use cases and technologies will be implemented during the project.

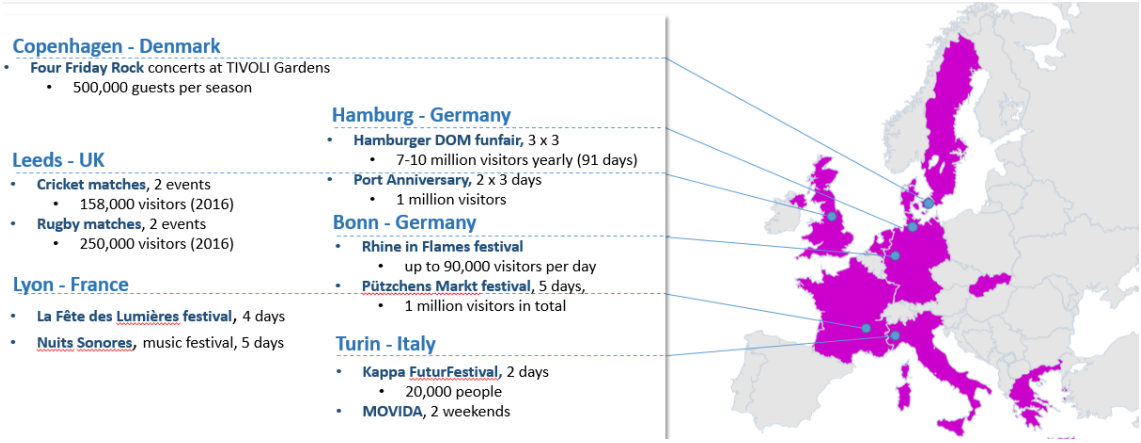


Figure 36: MONICA Pilots overview.

To agree which use cases are the most relevant across all pilots, a set of generic use cases have been prioritised according to the individual needs of the pilots. The technique used to achieve this result is based on the MoSCoW method, adapted to fulfil the objectives of MONICA project.

At the beginning, fifty generic high-level use cases have been defined and grouped in 12 categories, and the subsequent prioritisation by the pilots resulted in the selection of XXX use case groups for implementation.

The main use case group selected are the following:

- **Sound monitoring and control**: Copenhagen (DK), Lyon (FR), Bonn (DE), Torino (IT).
- **Crowd and capacity monitoring**: Copenhagen (DK), Lyon (FR), Bonn (DE), Leeds (UK), Torino (IT), Hamburg (DE)
- **Security - Health incidents**: Copenhagen (DK), Bonn (DE), Hamburg (DE), Torino (IT), Leeds (UK), Lyon (FR).
- **Missing persons/Locate staff members**: Copenhagen (DK), Hamburg (DE), Bonn (DE), Leeds (UK), Torino (IT), Lyon (FR).

MONICA architecture applies to of all Use Cases prioritised and selected. Based on pilot requests, in fact, some architectural modules are used and not others.

As an example, in LeedsFor the *ASFC Acoustica module* is not used to limit noise in the neighborhood. Some pilots use only UWB wearable, other pilots use only crowd wristbands (based on 868), and some pilots use both. In all these cases, the MONICA architecture structure remains the same.

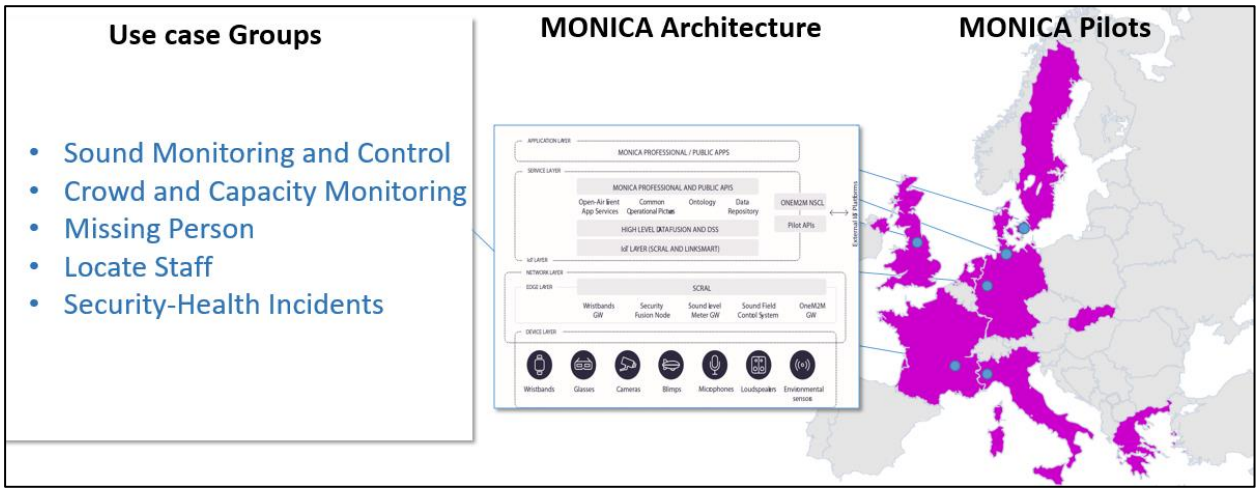


Figure 37: MONICA Use Cases and Architecture.

4.4.4 Benchmarking

MONICA approach to architecture can be summarized as presented in Table 7.

Table 7: MONICA summary

Number (and name) of reference standards used for the architectural study	AIOTI WG03 – IoT Standardisation, "High Level Architecture (HLA)", Release 3.0, June, 2017
Does it provide a minimum reference architecture?	Yes, as described in MONICA architecture paragraph
Is the project considering the use of common standards?	OneM2M, radio spectrum regulation for wristbands (ETSI EN 300 220-2 V3.1.1 (2017-02) and ETSI EN 302 065-2 V2.1.1 (2016-11)), OGC SensorThings, REST, RESTful.
Are use cases/demonstrators asked to comply with the common reference architecture?	Yes, all use cases are compliant with MONICA architecture.
Are there evaluation KPIs related to use case compliance with architecture? Please name them	No.

4.5 SYNCHRONICITY

Design of the SYNCHRONICITY architecture model started from the standard technologies and uniform analysis of relevant studies, coupled with a baseline of existing deployments in the partner cities. The analysis has been focused on finding commonalities of the similar works from the other studies.

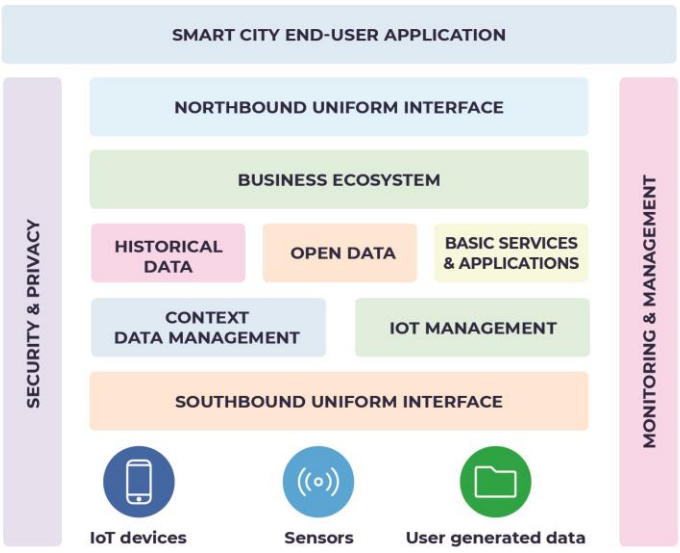


Figure 38: SYNCHRONICITY architecture simplified.

In order to make the in-depth comparison uniformed and straightforward, the analysis was conducted using a survey approach, by identifying the key points that showed the relevant aspects of core technologies and functionalities that underpin smart city platforms.

The identified commonalities were the starting point to provide a common SYNCHRONICITY framework for cities and they have been reflected into the design of the SYNCHRONICITY framework.

4.5.1 Reference architecture

SYNCHRONICITY conducted a first analysis on current existing architectures regarding different studies and initiatives, with special focus on current EU reference projects, such as, AIOTI, FIWARE, BIG-IoT among others.

Using this analysis as baseline, the working assumption for SYNCHRONICITY is that there is a consensus about a core layered architecture, as described Figure 39.

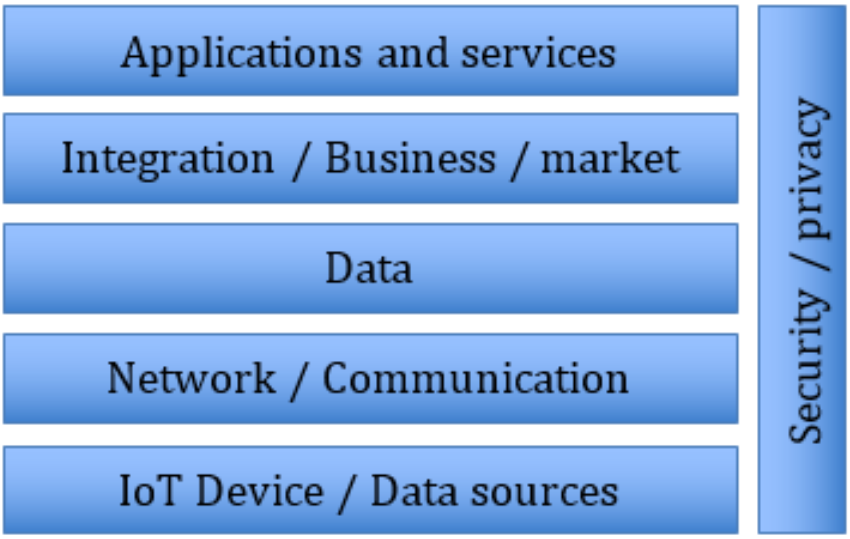


Figure 39: Synchronicity core architectural layers, as described in [19]

SYNCHRONICITY D2.1.1 deliverable on architecture [19] follows with an analysis of the current architectures deployed by the reference zones (pilots) of the project.

This analysis shows these architectures using a layered approach so as to ease the reference to the high-level architecture. Finally, a set of generic requirements are derived from the architectural analysis, considering also other previous deliverables in the project about principles, guidelines, data protection, privacy, etc.

These requirements, altogether, are used to produce the project-wide architecture as described in Figure 40. All in all, SYNCHRONICITY architecture is not purely based on a given and previous reference.

On the contrary, it is the result of the analysis of current architectures, pilots and project requirements, resulting into the creation of a customized vision for the project.

Nevertheless, the final proposal might be also considered as an instantiation of the generic (Figure 39) multi-layered architecture.

Finally, it is worth remarking that there is a key point about using previous or existing architectures, as SYNCHRONICITY fully supports OASC (Open & Agile Smart Cities) principles (see section 3.1.8) including:

- A common standard API for context information management (NGSI) and
- A common set of information models and finally a standard data publication platform

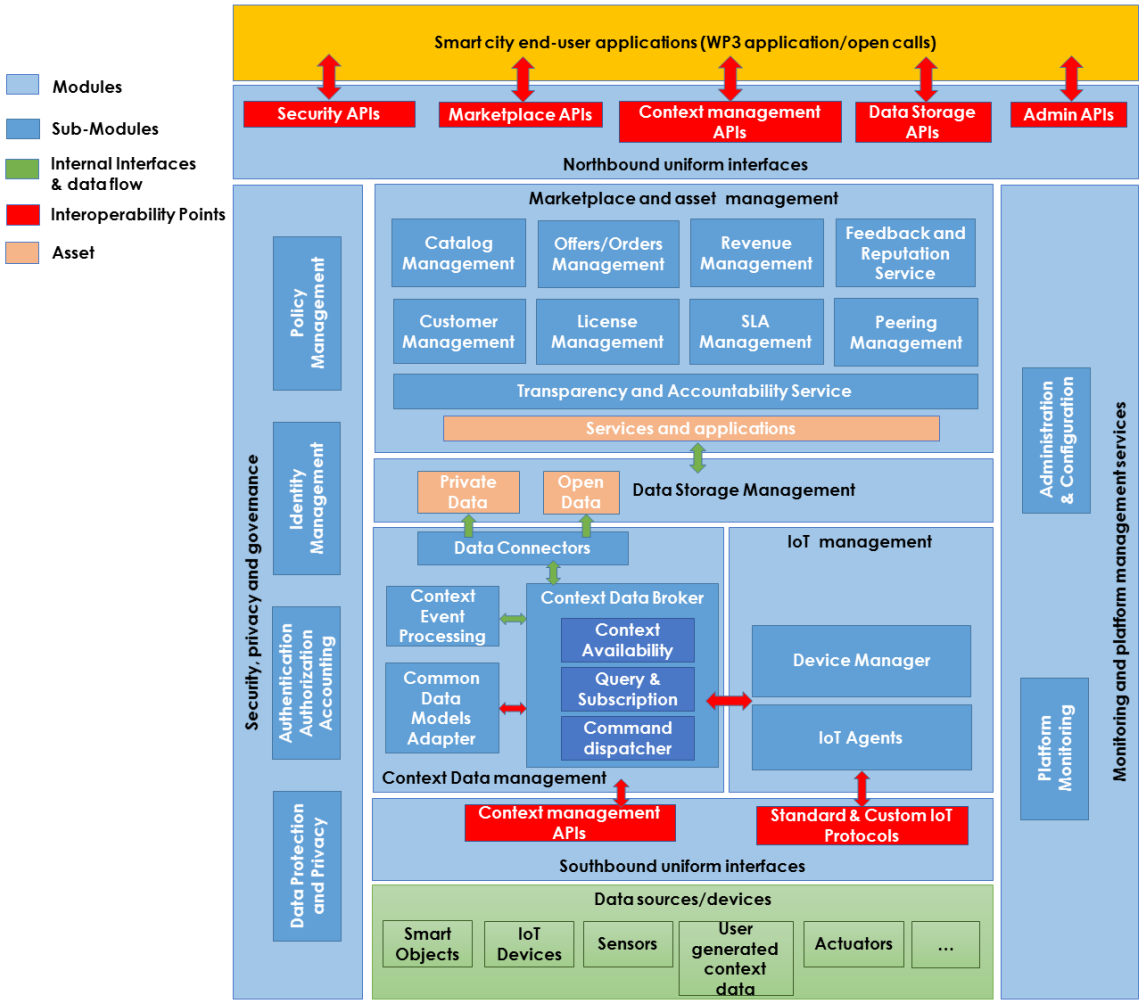


Figure 40: SYNCHRONICITY Reference Architecture extracted from [19]

4.5.2 Architectural focus

Being SYNCHRONICITY still in early deployment stage, assessing the reference architecture adoption on pilots/demonstrators/cities is not yet feasible, although there are expected to follow this baseline.

SYNCHRONICITY is by design covering "cross-domain use cases", and it aims to separate concerns of tight and loose couplings. This is done base on the emerging standards for OASC Minimal Interoperability Mechanisms, on which SYNCHRONICITY is explicitly founded, and it includes the work at ETSI and SF-SSCC, which takes input from initiatives like FIWARE, oneM2M, EIP-SCC etc. The technical vision is based on the four points (neutral branding based on standards, minimal interoperability mechanisms, reference architecture model and hosting options).

Adopting OASC principles as maximum premise will mainly affect different layers, especially regarding the use of NGSI interfaces and APIs. Therefore, data context and northbound APIs are core to the proper setup of SYNCHRONICITY demonstrators, and they should implement NGSI interfaces. In parallel, project partners are also working on the conversion of legacy datasets to data models based on NGSI standards.

As a result, each pilot is currently converting (or has already converted) their datasets using data models, or they are creating them from scratch using NGSI standards based on FIWARE data models.

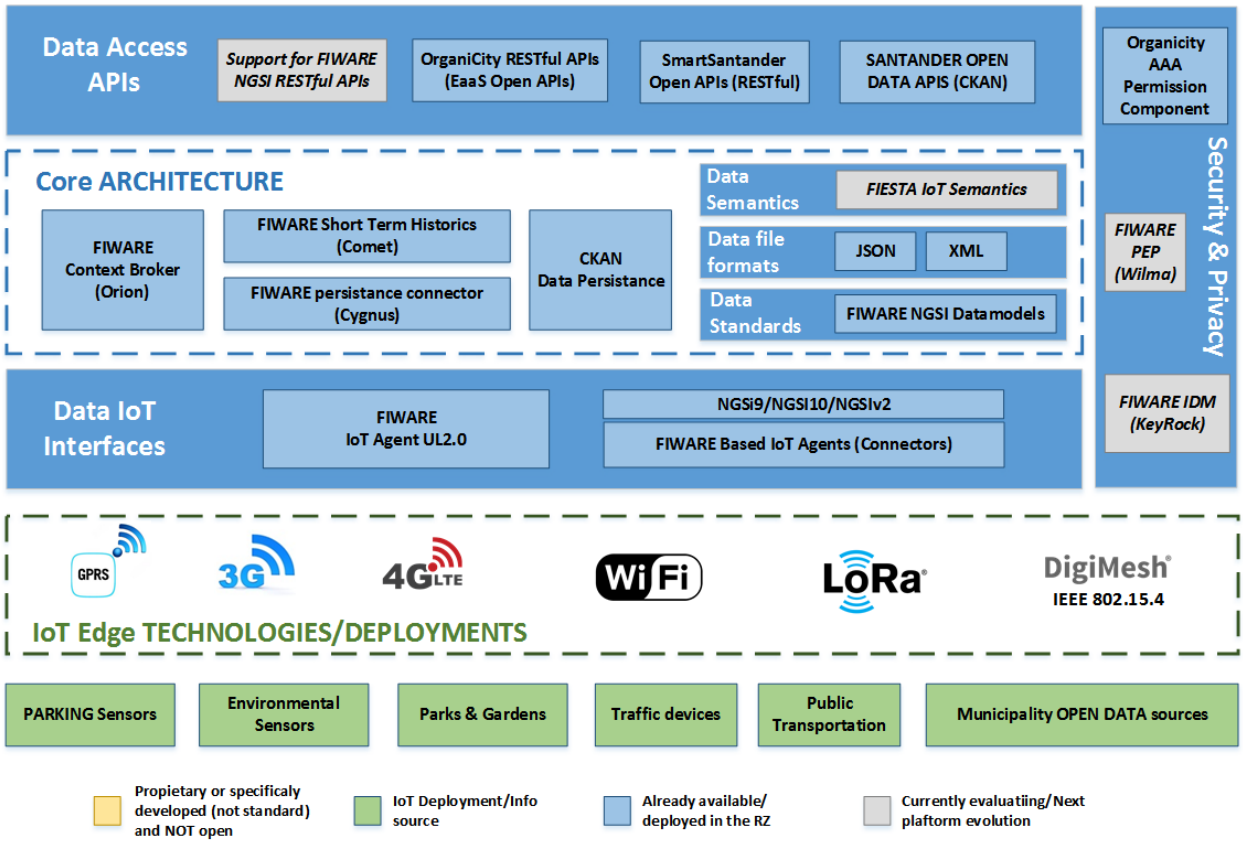


Figure 41: Example of current architecture of one of the pilots (Santander) already following some OASC principles [19]

In any case, all pilots have positioned their current situation according to the multi-layered generic architecture trying to concrete current logical components. Therefore, it is safe to assert that SYNCHRONICITY approach is architecture-oriented, being city pilots asked to adapt their deployments to the project-wide view presented. It is also fair to acknowledge that this approach was feasible as per the preliminary study in which all demonstrators were able to align their current status with respect to the minimum consensus layered architecture as illustrated in Figure 41.

4.5.3 Use Cases

The project defines three different use cases (known as application themes). Each of these themes is subscribed by different interested pilots. Therefore, each application theme is susceptible to be replicated in a maximum of 8 different cities, resulting on a potential landscape of 8x3=24 different use cases.

The application themes are summarized as:

- **Human-centric traffic management,** aiming at improving bicycle mobility in cities, making use of different IoT systems.
- **Multi-modal transportation**, implementing a mobile user application to provide multimodal routes (public transport, car sharing, etc) to citizens, making use of other mobiles and available data (tickets, timetables, etc…).
- **Community policy suite**, a city council service that, making use of different data streams, support the definition, implementation and evaluation of policies.

The pilots interest about the different applications themes can be summarized in the following table:

Application Theme	RZ
Human-centric traffic management	*Antwerp*
	Eindhoven
	Milan
Multimodal transportation	*Sandander*
	Porto
	Milan
	Helsinki
Community policy suite	*Manchester*
	Porto
	Carouge

Figure 42: Interests of Pilots(RZs) and application themes

As previously mentioned, these application themes can be deployed in a set of project city partners and will be constructed above the reference architecture. A set of common base-line services will be implemented exploiting the northbound layer facilitating the application themes implementations. On the other hand, pilot's data sources/streams, devices and sensors need to be integrated into the architecture through the southbound layer. Both cases, following the OASC principles, and therefore, implementing NGSI interfaces and data models that are currently under study phase.

SYNCHRONICITY focused first on the alignment for the architecture, producing a baseline layered approach of common use in all potential use cases. Now it is focusing on the use case particularization and working with pilots/cities to deploy and align homogeneous use cases across the cities.

4.5.4 Benchmarking

SYNCHRONICITY approach to architecture can be summarized as presented in Table 8.

Table 8: SYNCHRONICITY summary

Number (and name) of reference standards used for the architectural study	Smart City Reference Architectures: ITU-T FG-SSC [20], ITU-T SG13 Y.2060 [21], ISO/IEC JTC1[22], oneM2M [23]. Smart City Reference Architectures from EU projects: FIWARE [38], AIOTI [39], EIP-SCC [40], ESPRESSO [41], BIG-IOT [42], Organicity [43], Triangulum [44], symbIoTe [45].
Does it provide a minimum reference architecture?	Yes, minimum layered architecture plus extended common reference architecture
Is the project considering the use of common standards?	Yes, NGSI, OAUTH2, DCAT-AP, TM FORUM ecosystem API, ETSI NGSI-LD
Are use cases/demonstrators asked to comply with the common reference architecture?	Yes
Are there evaluation KPIs related to use case compliance with architecture? Please name them	None

5. DISCUSSIONS

At a first look, it might seem that the LSPs approaches are very different from one another. Nevertheless, this is due to the fact that they are produced by different teams with different needs and objectives, resulting in on the one hand architectural approaches with extended functionalities and detailed modules and on the other hand with architectural approaches providing high level recommendations or domain-specific implementations.

Digging a bit deeper on these approaches, it is possible to identify a number of common solutions adopted that are comparable and forming the basis for a common generic architectural approach. These elements are described in the following sections.

The analysis in the document aims to define a framework for the conceptualisation of LSPs' IoT architectures, supporting the understanding of the IoT pilot's essence and key properties pertaining to its behaviour, composition and evolution, which can affect the feasibility, utility, maintainability and sustainability of the IoT large-scale pilots.

The analysis shows that the different projects have used different architectural descriptions based on the work presented in standardisation bodies and organisations such as: ITU-T SG13 Y.2060, ISO/IEC JTC1, oneM2M, FIWARE/ETSI. The architectural views expressed by the projects represent in general the architecture of the pilot in accordance with different IoT architectural viewpoints. The architectural description of each LSP includes in different degrees, the 8 layers of the generic IoT architecture presented in Figure 19, while using different vocabulary and terminology.

The different LSPs' architectural views are integrated, layered structures that carry value-laden data from various IoT devices to networks, processing, abstraction and application layers to deliver implementations using various types of IoT platforms.

The software platforms used in the application layer are suited to delivering the key components for implementing various IoT applications that connect users, business partners, devices, machines and enterprise systems with each other. The information interpretation varies across the LSPs and relates to the specific requirements for each application domain.

One interesting observation is that the IoT platforms in some cases are included as part of a specific layer in the architecture, which suggests that their features are used primarily to address the functions required by that specific architectural layer.

For all the architectural approaches used by LSPs, the application layer interacts with the service layer, while many software applications are based on vertical markets/domains covered by LSPs, which defines the nature of device data and business needs.

In general, all the architectural approaches can be mapped to a high-level architecture consisting of 4 layers: one layer of things/devices, with sensing/actuating features, heterogenous complexity and autonomous capabilities (i.e., energy, functions, mobility, etc.); a layer including distributed intelligence processing, data aggregation processing, storage and networking; a layer for edge processing, data sharing infrastructure, information networking analytics and services; and a layer for applications, management, data centre infrastructure, cloud software infrastructure and back-end data centre systems.

All five LSP projects provide minimum reference architecture and use the reference architectures promoted by standardisation bodies and organisations such as: oneM2M (2), ITU-T SG13 Y.2060 (3), ISO/IEC JTC1 (2), FIWARE (2), AIOTI HLA (2).

Two LSP projects have several evaluation KPIs related to use case/demonstrator compliance with IoT architectures, one project does not yet have evaluation KPIs related to use case/demonstrator

compliance with IoT architectures, and two other projects do not have evaluation KPIs related to use case/demonstrator compliance with IoT architectures.

The analysis and mapping of pilot architecture approaches are important steps in addressing IoT implementation across the pilots, and for providing critical requirements for upcoming IoT architectures.

This supports the validation of technology deployment, replicability towards operational deployment, and gap identification for IoT architecture regarding interoperability and standards approaches at technical/semantic levels to address scalability, interoperability, end-to-end security and Quality of Service (QoS).

The results of the work are used for the mapping of pilot architecture approaches to different use cases for describing local architectures (FIWARE, oneM2M, etc.) and integration into a generic IoT architecture framework.

Further work focusses on definition of a common generic IoT reference architectures that can be adapted to the different domains covered by the LSPs and to convey a common approach to structuring IoT deployments. Generic IoT reference architectures can be fine-tuned for a specific use case or application if needed.

The work will continue in Activity Group 02 and evolve further through a set of recommendations and best practices that will form the basis for discussions on pre-normative and standardisation activities across the application domains. The work will further concentrate on promoting the commonalities and best practices of standardisation to extend the functionality of open standards for future IoT applications across vertical domains. Moreover, while the expansion of IoT is growing through the adoption of new applications, future IoT architectures must meet scalability, interoperability and end-to-end security requirements.

6. REFERENCES

[1] The AUTOPILOT Project, (AUTOmated driving Progressed by Internet Of Things). Online at: http://autopilot-project.eu/

[2] oneM2M Technical Specification. *Functional Architecture*, TS-0001-V2.10.0, 30-08-2016. Online at: http://www.onem2m.org/images/files/deliverables/Release2/TS-0001-%20Functional_Architecture-V2_10_0.pdf

[3] *Report on development and Integration of IoT devices into IoT ecosystem*. AUTOPILOT D2.4, June 2018.

[4] *Performance and KPIs for autonomous vehicles and IoT pilot impact measurement*. AUTOPILOT D5.3, December 2017.

[5] IEEE Project 2413 - Standard for an Architectural Framework for the Internet of Things (IoT), online at: https://standards.ieee.org/develop/project/2413.html

[6] ISO/IEC/IEEE 42010 Systems and software engineering - Architecture description, online at: http://cabibbo.dia.uniroma3.it/asw/altrui/iso-iec-ieee-42010-2011.pdf

[7] ISO/IEC CD 30141, Internet of Things Reference Architecture (IoT RA), online at: https://www.iso.org/standard/65695.html

[8] oneM2M Architecture, online at: http://www.onem2m.org/application-developer-guide/architecture

[9] *FI-WARE NGSI-9 Open RESTful API Specification*. FIWARE Foundation 2017. Online at: https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/FI-WARE_NGSI-9_Open_RESTful_API_Specification

[10] *FI-WARE NGSI-10 Open RESTful API Specification*. FIWARE Foundation 2017. Online at: https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/FI-WARE_NGSI-10_Open_RESTful_API_Specification

[11] *AIOTI - High Level Architecture (HLA)*, rel. 3.0. AIOTI WG03-IoT Standardisation, June 2017. Online at: https://aioti.eu/wp-content/uploads/2017/06/AIOTI-HLA-R3-June-2017.pdf

[12] Internet of Things Architecture (IoT-A). European Commission, Community research and Development Information Service (CORDIS). Online at: https://cordis.europa.eu/project/rcn/95713_en.html

[13] IoT-A. *Final architectural reference model for the IoT*, v3.0 (D1.5). Online at: https://iotforum.org/wp-content/uploads/2014/09/D1.5-20130715-VERYFINAL.pdf

[14] https://www.fiware.org/about-us/

[15] ETSI Industry Specification Group for cross-cutting Context Information Management: http://www.etsi.org/deliver/etsi_gs/CIM/001_099/004/01.01.01_60/gs_CIM004v010101p.pdf

[16] FIWARE Context Broker GitHub repository: https://github.com/telefonicaid/fiware-orion

[17] FIWARE GE catalogue : https://catalogue-server.fiware.org/

[18] FIWARE data models: https://www.fiware.org/developers/data-models/

[19] Reference Architecture for IoT Enabled Smart Cities SYNCHRONICITY D2.1.1 deliverable. August 2017.

[20] ITU-T Focus Group on Smart Sustainable Cities. https://www.itu.int/en/ITU-D/Regional-Presence/AsiaPacific/Documents/FG-SSC_generic_Cristina_Bueti_r2%20(19%20september%202014).pdf

[21] ITU-T Y.2060 : Overview of the Internet of things. https://www.itu.int/rec/T-REC-Y.2060-201206-I.
[22] ISO/IEC JTC 1 — Information Technology. https://www.iso.org/isoiec-jtc-1.html
[23] OneM2M http://www.onem2m.org/.
[24] IoT Open Platforms. *IoT-A Architectural Reference Model*. Online at: http://open-platforms.eu/standard_protocol/iot-a-architectural-reference-model/
[25] IoT Forum organization. Online at: https://iotforum.org/
[26] E. Gazis, et.al. IoT: Challenges, Projects, Architectures. 18th International Conference on Intelligence in Next Generation Networks, February 2015. Online at: https://www.researchgate.net/publication/273126211_IoT_Challenges_Projects_Architectures
[27] International Organization for Standardization (ISO). *ISO/IEC CD 30141 - Internet of Things Reference Architecture (IoT RA)*, ISO 2016 Online at: https://www.iso.org/standard/65695.html
[28] ITU-T, Smart Sustainable Cities at a Glance. Online at: https://www.itu.int/en/ITU-T/ssc/Pages/info-ssc.aspx
[29] *Shaping smarter and more sustainable cities - Striving for sustainable development goals*. ITU-T's Technical Reports and Specifications, 2016. Online at: https://www.itu.int/wftp3/pub/epub_shared/TSB/ITUT-Tech-Report-Specs/2016/en/flipviewerxpress.html
[30] *Internet of things (IoT) and smart cities and communities (SC&C)*. Online at: https://www.itu.int/en/ITU-T/studygroups/2017-2020/20/Pages/default.aspx
[31] ITU-T SG13 Future networks focus on IMT-2020 cloud computing and trusted network infrastructure. Online at: https://www.itu.int/en/ITU-T/studygroups/2017-2020/13/Pages/default.aspx
[32] ITU-T Y.2060: Next Generation Networks - Frameworks and functional architecture models, June 2012.
[33] ITU-T Y.4414/H.623: Web of things service architecture (Internet of things and smart cities and communities - Frameworks, architectures and protocols / Broadband, triple-play and advanced multimedia services - Advanced multimedia services and applications), November 2015.
[34] PAS 182 Smart city concept model. Online at: https://www.bsigroup.com/en-GB/smart-cities/Smart-Cities-Standards-and-Publication/PAS-182-smart-cities-data-concept-model/
[35] OASC Background document: http://www.oascities.org/wp-content/uploads/2015/11/Copy-of-Open-and-Agile-Smart-Cities-Background-Document-6th-Wave.docx.pdf
[36] Open & Agile Smart Cities web page: http://oascities.org/about-oasc/
[37] CitySDK project web page: http://www.citysdk.eu/
[38] FIWARE Foundation. Online at: https://www.fiware.org/
[39] Alliance for Internet of Things Innovation (AIOTI). Online at: https://aioti.eu/
[40] The Marketplace of the European Innovation Partnership on Smart Cities and Communities (EIP-SCC). Online at: https://eu-smartcities.eu/
[41] Systemic standardisation approach to empower smart cities and communities (ESPESSO). Online at: http://espresso-project.eu/
[42] Bridging the Interoperability Gap of the Internet of Things (BIG-IoT). Online at: http://big-iot.eu/
[43] The OrganiCity project. Online at: http://organicity.eu/
[44] The Triangulum project. Online at: http://triangulum-project.eu/
[45] Symbiosis of smart objects across IoT environments (symbIoTe). Online at: https://www.symbiote-h2020.eu/
[46] Web of Things Working Group. Online at: https://www.w3.org/WoT/WG/

[47] The Eclipse Thingweb project. Online at: **Error! Hyperlink reference not valid.**projects.eclipse.org/proposals/eclipse-thingweb
[48] Web of Things (WoT) Architecture. Online at: https://www.w3.org/TR/wot-architecture/
[49] ISO/IEC 11179 Information Technology – Metadata registries (MDR) – Part 1: Framework https://www.iso.org/obp/ui/#iso:std:iso-iec:11179:-1:ed-3:v1:en
[50] ISO 15000-5:2014 Electronic Business Extensible Markup Language (ebXML) https://www.iso.org/standard/61433.html
[51] ITU-T Focus Group on Smart Sustainable Cities – https://www.itu.int/en/ITU-T/focusgroups/ssc/Pages/default.aspx
[52] ITU-T Study Group SG20: Internet of things (IoT) and smart cities and communities (SC&C) https://www.itu.int/en/ITU-T/studygroups/2017-2020/20/Pages/default.aspx
[53] AIOTI WG03 Reports. *AIOTI WG03 Reports on IoT Standards*, June 2017. Online at: https://aioti.eu/aioti-wg03-reports-on-iot-standards/