H2020 – LC – SC3 – EE – 2019 – GA 894240

Operating System for Smart Services in Buildings

D2.1 Report on Requirement Analysis for IoT Ecosystems

WP2 IoT for Smart Buildings

	Name	Date
Prepared by	Frédéric Revaz (HES-SO), Dominique Gabioud (HES-SO)	15.01.2021
Peer reviewed by	Sašo Brus (INEA DOO), Amir Laadhar (AAU)	16.02.2021
Reviewed and approved by	Dominique Gabioud (HES-SO)	28.02.2021

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 894240.

Distribution list

External		Internal	
European Commission	x	Consortium partners	x

Change log

Version	Date	Remark / changes
1.1	15.01.2021	First version by HES-SO
2.1	22.02.2021	Second version by HES-SO
3.1	28.02.2021	Final version by HES-SO

To be cited as

"D2.1 Report on Requirement Analysis for IoT Ecosystems" of the HORIZON 2020 project domOS, EC Grant Agreement No 894240.

Disclaimer

Table of contents

Executive summary **5**

1. Introduction **5**
1.1. Document structure 5
1.2. IoT and Existing Smart Buildings 5
1.2.1. Overview of Current Status 5
1.2.2. Accelerators for IoT in Existing Buildings 6
1.2.3. Barriers for IoT in Existing Buildings 6
1.2.4. Communication Infrastructure 6
1.3. Scope of the Project 7
1.4. Definitions 7
1.5. Objectives and Success Criteria 8
1.6. Work structure 9
1.7. Implementation of the IoT Ecosystem on the Three IoT Platforms 10

2. Proposed IoT for Smart Buildings **10**
2.1. Overview 10
2.1.1. Methodology 10
2.1.2. Overview of the IoT Ecosystem 10
2.2. Functional Requirements 11
2.2.1. Requirements collection 11
2.2.2. Requirements for Platforms 11
2.2.3. Requirement for Applications 12
2.2.4. Requirement for Smart Systems 12
2.2.5. Requirements on Semantics 13
2.2.6. Requirements on Deployment Topology 13
2.2.7. Priority 14
2.3. Non-Functional Requirements 15
2.3.1. Compliance with Recognized IoT / Web Standards 15
2.3.2. Requirements on Privacy 16
2.3.3. Security Requirements 17
2.3.4. Safety 17
2.3.5. Usability 17
2.3.6. Performance 17

3. Scenarios **18**
3.1. Introduction 18
3.2. Smart System Description 18
3.3. Smart Systems Directory 18
3.4. Subscription to an Application 18
3.5. Intermediary Function 19

4. Conclusion **21**

5. References **21**

List of figures

Figure 1: Elements in the IoT Ecosystem..... 11
Figure 2: Two Possible Topologies: (a) Distributed Cloud Solution (B) Edge Solution 14
Figure 3: Illustration of the Subscription Process 19
Figure 4: Illustration of the Intermediary Concept in the WoT Architecture [Wot Architecture] (Consumer: Application in the Present Document, Thing: Smart System, Thing description: Smart System Description 20
Figure 5: Implementation of the Intermediary Based on the Servient Concept [WoT Architecture]..... 20

List of tables

Table 1: Definitions 7
Table 2: Success Criteria.....8
Table 3: Priority in Requirements 14
Table 4: Status of W3C WoT Standards 15

Terms, definitions, and abbreviated terms

GA	Grant Agreement
ICT	Information and Communication Technology
IETF	Internet Engineering Task Force

Executive summary

Adapting older buildings to make them intelligent using IoT technologies is made difficult by several barriers: the long lifespan of building-related devices, the cost of retrofitting older installations, and the tendency for these systems to operate in silos. domOS seeks to reduce, if not remove some of these barriers by offering a standardization of IoT layers in the building industry. The domOS vision is to make it possible for any smart application to use of any smart infrastructure, independently of its technology.

In this work package, the focus is on two areas: firstly, research into the technical possibilities for making energy systems smarter, with an emphasis on older systems, and secondly, the search for a suitable infrastructure for establishing an interoperable ecosystem for buildings.

By introducing the concepts of application, platform, container, and smart-system and assigning precise roles to each of these entities, it is possible to achieve this goal.

This document defines the actors of the ecosystem, specifies their role and requirements through different scenarios. This document also explains how the actors of this ecosystem must collaborate. Functional and non-functional needs are expressed.

1. Introduction

1.1. Document structure

This document is divided into three parts:

- The first part of this document describes the current state, the stakes of the project and above all defines the different elements of the IoT ecosystem that is envisaged.
- The second part describes the IoT ecosystem and defines the functional and non-functional requirements. At the end of the second part, the priorities in the functional requirements are outlined.
- The third part shows different scenarios in order to specify the implementation of the system.

1.2. IoT and Existing Smart Buildings

1.2.1. Overview of Current Status

The expectations for smart buildings are high and cover a variety of areas: increase energy efficiency, empower the building's occupants, raise comfort and security levels, better integrate buildings into energy grids. Smart services require sensing, actuating, communication, user interface, and intelligence. The Internet of Things (IoT), which covers the three first requirements, is indeed an enabler for smart services.

Compared to other sectors, the degree of digitalisation of buildings in Europe and elsewhere can be qualified as low:

- The deployment of building automation solutions is mostly limited to large and/or high-end residential or service buildings.

- As energy appliances (e.g., boiler) have a long lifespan, a large part of them date from the pre-digital area and therefore do not feature any digital interface. Integrating such appliances in IoT is a key issue to increase the energy efficiency and flexibility.
- Offers for islanded smart systems (from smart coffee machines to Wi-Fi heat pumps) flourish. Integration – at the building or household level – of several smart systems is complex (i.e., resource consuming) and sometimes impossible.

1.2.2. Accelerators for IoT in Existing Buildings

The primary driver is the wish of citizens for better control of their building. Understanding the citizens' expectations is of course essential, but it is not the scope of the present document.

Technologies – mainly microelectronics, telecommunication, and informatics – have made huge progresses over the last decade:

- Cost of adding connectivity to any microprocessor-based system is low and represents a small proportion of the system cost, even for simple systems.
- Internet connectivity is ubiquitous. Price per bit tends toward zero.
- Integration of connected devices / appliances with the web has become a mature technology.

1.2.3. Barriers for IoT in Existing Buildings

There are at least four main barriers to adopting IoT in existing buildings:

1. Current IoT solutions deployed in buildings work as silo systems: each one has its own infrastructure, users must cope with multiple access schemes and user interfaces, "cross-fertilisation" between devices / appliances is not possible (closed systems) or complex (multiple APIs).
2. Domestic appliances can have a long lifespan (30 years or more). Many installed appliances do not feature any digital interface and no telecom cabling infrastructure has been installed in their surroundings.
3. Integrated building automation solutions are available on the market, but most of them require expensive custom projects, and are therefore not appropriate for the deployment in existing buildings.
4. Companies that build domestic appliances fear obsolescence and struggle to integrate intelligent functions.

1.2.4. Communication Infrastructure

The communication infrastructure of existing buildings in Europe ranges from no infrastructure at all to full universal cabling with several telecom sockets per room.

Existing cabling systems can of course ease the deployment of smart solutions, but installing ad hoc cabling is difficult in existing buildings, for cost and aesthetic reasons. Therefore, IoT solutions for existing buildings shall provide a fully wireless option.

As internet infrastructure is expanding rapidly in Europe, high-speed internet access is assumed to be available for all buildings.

1.3. Scope of the Project

WP2 "IoT for Smart Buildings" addresses two specific areas of smart buildings:

1. The question of the equipment of existing buildings, either residential or tertiary, with systems for smart energy services is addressed. Finding a way to make old equipment smarter is a priority but is not mandatory. Relevant processes and appliances are identified, appropriate ways to interface them for monitoring and control are elaborated. In-door communication systems are assessed.
2. The question of the interoperability of smart building solutions is addressed. The vision is that any smart Application can make use of any smart infrastructure, independently of its technology. This will boost the development of smart applications as they can have access at a large building basis, potentially without investing in hardware appliances, devices, or communication gateways.

1.4. Definitions

Table 1: Definitions presents definitions used throughout WP2.

TABLE 1: DEFINITIONS

Topic	Definition	Example
Connected System	Digital system in a **Container** providing a data interface for an external information system	A Wi-Fi heat pump, a SunSpec compatible solar inverter, a Zaptec charging station for electrical vehicle, a Z-Wave controller with its wireless peripherals
Smart System	**Connected System** made compliant with the **domOS IoT Ecosystem**. A Smart System is affected to only one building.	A connected system together with a cloud hosted domOS adapter component.
Customer	Natural (or legal) person in charge of the management of a **Container** and using a **Service**.	An adult inhabitant in a household, a facility operation in a service building or in a multi-family residential building
Container	In this context, premises where the collection of **Smart Systems** managed by a **Customer** are located: an apartment in a multi-family residential building, a single-family house, the communal area in a building…	The single-family house where the four above mentioned Smart Systems are located.
Application	Software component interacting with one or more **Smart Systems** located in a **Building**	A component aiming at maximising the self-consumption of solar power by the heat pump and the electrical vehicle charging station
Service	Interplay of one or several **Applications** and of one or several	Self-consumption optimisation

	Smart Systems, generating an added value for a **Customer**.	
Platform	Software environment acting as a mediation entity between on one side **Containers** and their **Smart Systems** and on the side **Applications.**	A cloud hosted utility enabling self-consumption optimisation Application to interact with the building hosting a solar inverter, a heat pump and an electrical vehicle charging station.
domOS IoT Ecosystem	ICT architecture specifying the interaction of **Applications**, **Platforms** and **Smart Systems**.	

1.5. Objectives and Success Criteria

The WP2 objectives and their corresponding success criteria are presented in Table 2: Success Criteria.

TABLE 2: SUCCESS CRITERIA

Objective	Success criteria
To elaborate and prototype a standard-based architecture named "IoT Ecosystem", allowing: • **a decoupling between Smart Systems and Applications,**	1. An Application can interact with any Container, providing Smart Systems make the appropriate monitoring and control parameters available. 2. Smart Systems provide a description of: o their monitoring and control parameters using the domOS core ontology, o the access protocol(s) along with the security credentials, and of o the syntax of exchanged messages. 3. An Application can verify whether the Container is equipped with the required monitoring and control parameters before deployment. 4. The IoT Ecosystem is based on open standards promoted by recognised standardisation bodies.
• **owners to manage their privacy, and**	1. Owners explicitly allow an Application to operate with their Smart Systems. 2. Owners explicitly allow Applications to monitor/ control individual parameters. 3. Platforms provide Owners with an interface where they can centrally manage their privacy, and revoke rights given to Applications.
• **secure operation of buildings.**	1. Applications dispose of their own security credentials to access the Platform. 2. The Platform disposes of the security credentials to access Smart Systems.
To upgrade the three participating IoT Platforms (S-IOT, cloud.iO, ArrowHead) according to the defined IoT Ecosystem.	1. Each participating Platform provides an implementation of the IoT Ecosystem.
To prototype a smart Application capable of running over the three enhanced IoT Platforms.	1. A Proof of Concept (PoC) illustrates that a single prototype Application is capable to operate over each IoT Platform. Each Platform connects Smart Systems providing the same monitoring and control functions (e.g., electrical power

	monitoring and power supply control). The PoC will be validated by executing the 4 scenario presented in section 3.
To define a sound concept for retrofitting existing buildings with monitoring and control infrastructure for energy appliances.	1. The concept allows to retrofit all generations of buildings throughout Europe. 2. Monitored and controlled parameters enable smart services for energy efficiency and energy flexibility. 3. The overall cost (hardware, manpower) for installation is in-line with the expected benefits. Target values for Western EU : 100 € for a communication gateway, 1 hour work time onsite and 100 € hardware cost per connected appliance.

1.6. Work structure

WP2 shall provide results in two areas:

- Area 1: Elaborate solutions for connection of energy appliances in existing buildings, in order to be able to deploy smart services for energy efficiency and flexibility at large scale in Europe.
- Area 2: Elaborate solutions promoting interoperability between Applications and Smart Systems (IoT Ecosystem), so that:
    - Customers have an added value for their Smart Systems, as they can subscribe to more Services, and
    - Service providers have an added value for their Services, as they have a larger potential Customer basis.

Area 1 is handled in the task T2.2 “In-Building Infrastructure for Smart Services” and reported in D2.2 “Infrastructure for Smart Services” at month M12.

Regarding area 2, WP2 will specify an appropriate Ecosystem (domOS IoT Ecosystem), upgrade the three participating IoT Platforms, make them compliant with the IoT Ecosystem and develop a test and validation prototype.

The domOS IoT Ecosystem will be elaborated in the frame of the task T2.3 “Functional Specification for the IoT Ecosystem” and described in D2.3 “Functional Specification of the IoT-Ecosystem” at month M18.

The three participating Platforms will be adapted in tasks T2.4 “SIOT Platform Development”, T2.5 “ArrowHead Platform Application” and T2.6 “cloud.IO Platform Development”. Upgraded Platforms will be described in the deliverables D2.4 “Upgraded SIOT Platform”, D2.5 “Upgraded Arrowhead Platform”, and D2.6 “Upgraded cloud.iO Platform”.

In task T2.7 “Platforms Test and Validation”, a distributed testbed integrating the three Platforms will be developed. Test and validation results will be reported in deliverable D2.7 “Report from Test and Validation of the IoT Platforms” at month M30.

1.7. Implementation of the IoT Ecosystem on the Three IoT Platforms

During the requirement analysis phase, the work force has concentrated on gathering requirements for a commonly agreed architecture for IoT in smart buildings (the so-called domOS IoT Ecosystem). These requirements are presented in the present document.

At this stage, no estimation of the resources needed to upgrade the three platforms have been done. As the requirements are ambitious and based on emerging standards, the probability is high that resources will not be enough to cover the full implementation of the IoT Ecosystem on the three platforms.

In this context, a particular attention should be paid during the specification phase to identify key elements of the IoT Ecosystem whose implementation must be prioritized. Priorities should be defined according to the following criteria:

- needs of demonstrators,
- contribution to the implementation of the IoT Ecosystem.

The relevance of a common development of some components of the IoT Ecosystem should be evaluated even if it is not foreseen in the Description of Action.

2. Proposed IoT for Smart Buildings

2.1. Overview

2.1.1. Methodology

Decoupling Applications and Smart Systems is a requirement met in several domains – and not only in the smart building domain. Hence, it would not make sense that WP2 develops its own solution for this requirement (incidentally, WP2 would not have the resources for it). Consequently, domOS intends to base its IoT Ecosystem on recognized standards. domOS should nevertheless consider that the standardization landscape is evolving, and that the elaboration of new standards is underway. Therefore, the domOS IoT Ecosystem will consider established and emerging standards (see Section 2.3.1).

2.1.2. Overview of the IoT Ecosystem

For simplicity and in the first iteration of this IoT, the following hypothesis are assumed:

- a Customer is assumed to manage exactly one Container
- a Smart System is assumed to register to exactly one Platform
- A customer is registered to only one Platform.

In future evolution of this document, these hypotheses could be changed allowing more flexibility.

The central element of a domOS Ecosystem is the Platform. Once they have committed to a Platform, Customers can register one or more Smart Systems on it (black arrows).

Customers can subscribe to Applications (dashed arrows). Applications can only be activated (black lines) if the Smart Systems in the corresponding Containers provide appropriate monitoring and control parameters, and if it is compliant with privacy rules defined by Customers.

An overview of the IoT Ecosystem is provided in Figure 1: Elements in the IoT Ecosystem.

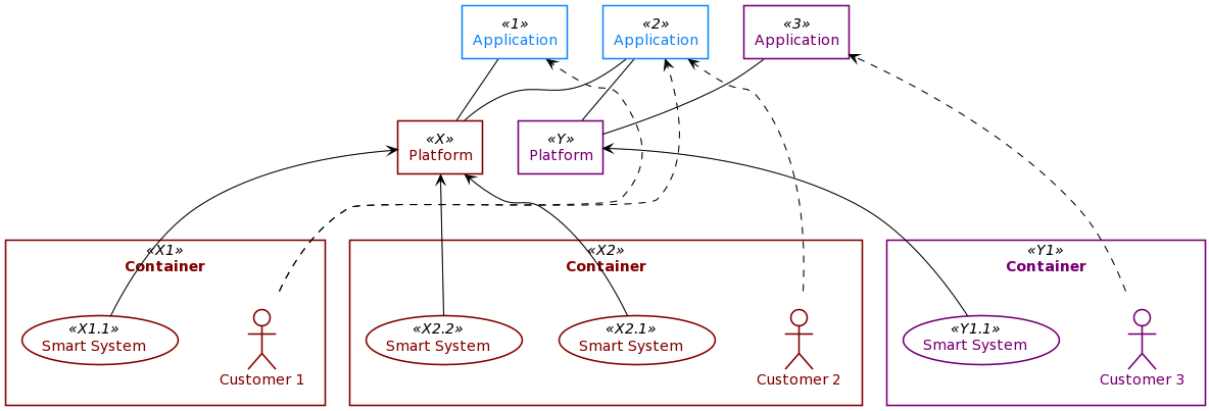


FIGURE 1: ELEMENTS IN THE IOT ECOSYSTEM

Services are the combination of Smart Systems and Applications to add value to a Customer. In figure 1, only one service is represented in purple: Customer 3 subscribes to Application 3 that operates on Smart System Y1.1 through Platform Y. Another service may be the combination of Application 1 and 2, to which Customer 1 has subscribed.

Compliant Platforms implementing the IoT Ecosystem can be many, possibly based on different implementations. A Container is hooked to only one Platform. On the contrary, Applications can access multiple Platforms.

2.2. Functional Requirements

2.2.1. Requirements collection

The requirements have been collected with the following methods:

- Brainstorming and plenum sessions
- Setting up of user stories
- Simulations and scenarios.

2.2.2. Requirements for Platforms

A Platform is a compulsory mediation point between Applications and Smart Systems, i.e., there are no direct links between Applications and Smart Systems.

A Platform is a pure ICT player. As such, it does not care about the business aspects of smart services.

A Platform shall:

- have access to a description of the Smart Systems hosted by participating Containers [FRP01],

- allow a registered Application to verify whether a Container disposes of the appropriate infrastructure for the service [FRP02],
- act as an intermediary between Applications and Smart Systems.

In its intermediary function, the Platform shall:

1. handover messages from Applications to Smart Systems, and from Smart Systems to Applications [FRP03],
2. accept secured connection from registered Applications [FRP04],
3. implement secure connections to registered Smart Systems [FRP05], and
4. implements privacy rules defined by the Customers [FRP06].

At most one Application shall control a set point in a Smart System [FRP07]. This rule shall be enforced by a Platform on a "first come – first serve" basis. Interferences between multiple services accessing multiple actuators (e.g., one Service turns on the heater and one other Service opens a window) shall be managed at the Service level.

2.2.3. Requirement for Applications

An Application is a software component that interacts with Smart Systems. This interaction may constitute an entire service, a part of a service or a part of multiple services.

Before activation, an Application shall verify that Smart Systems inside a Container offer the required functions [FRA01].

From a service perspective, Applications interact directly with Smart Systems.

From a technical perspective, Applications connect to a Platform. An Application shall use a unique set of security credentials defined by the Platform, i.e., it does not share any security credentials with Smart Systems.

2.2.4. Requirement for Smart Systems

Containers host one or more connected system(s) featuring internet connectivity.

Smart Systems are connected systems made compliant with the domOS Ecosystem. Turning a connected system into a Smart System should only require the provision of a description of the connected system (i.e., no protocol adaptation, no message translation, no modified security scheme, or security credentials).

Considering that a Platform is the peer communication entity of a Smart System, and that Platforms can support a limited set of protocols, the integration of systems only through description is only possible if the connected systems implement protocols, message formats and access control schemes supported by Platforms. The legacy technologies supported by Platforms shall be defined. In this document, only systems whose communication is supported by Platforms are considered. Other systems require an Application-level gateway.

A Smart System shall register on one and only one Platform. As part of the registration process, it makes its description available to the Platform [FRS01].

The component turning a connected system into a Smart System can be implemented either on a local gateway in the Container premises, as a cloud service, or natively in the connected system.

A Smart System description can be either static (i.e., typically provided in a text file), if the Smart System configuration remains stable over time, or dynamic (i.e., generated from a Smart System internal directory).

2.2.5. Requirements on Semantics

WP3 “Common Ontology and Semantics” develops the domOS core ontology, which will define naming conventions for relevant concepts in buildings.

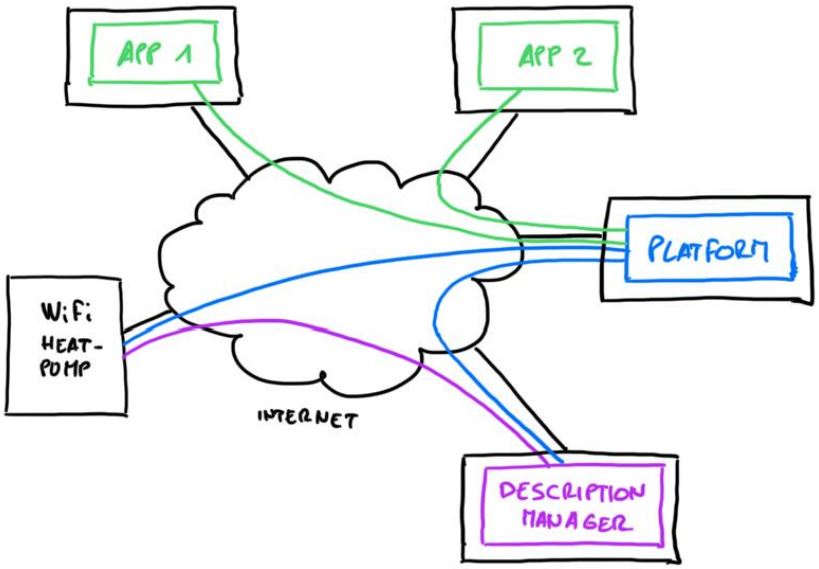
Smart Systems descriptions should associate domOS core ontology elements with information enabling the concrete remote (read and/or write) access to current element values in the Smart System [FRS02].

2.2.6. Requirements on Deployment Topology

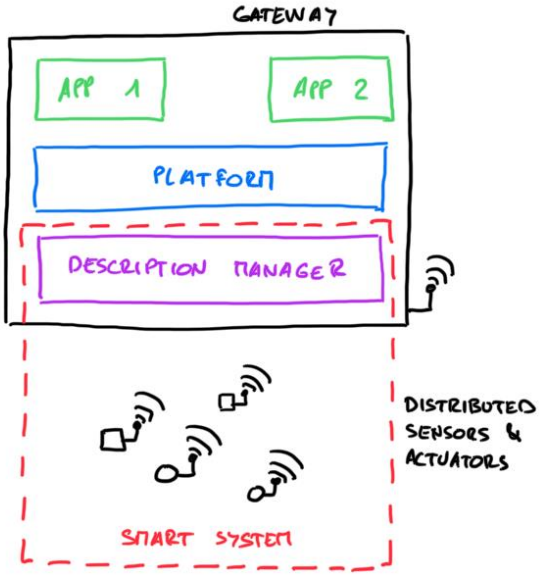
Connected systems are physical devices / appliances installed in the Customer premises.

A Platform and an Application are components that can be deployed using different topologies.

It should be possible to use edge and cloud hosting at all levels. Two possible topologies are presented in Figure 2: Two Possible Topologies: (a) Distributed Cloud Solution (B) Edge Solution.



(a)



(b)

FIGURE 2: TWO POSSIBLE TOPOLOGIES: (A) DISTRIBUTED CLOUD SOLUTION (B) EDGE SOLUTION

The choice of a topology appropriate for a given context is outside the scope of the WP2.

The IoT Ecosystem should put as few constraints as possible on underlying topologies.

IoT Platforms may not support all possible topologies.

2.2.7. Priority

Table 3 presents the priorities in the functional requirements. The most important requirements are presented with the smallest number.

TABLE 3: PRIORITY IN REQUIREMENTS

Priority	Functional requirement identifier and justification
1	[FRP01] [FRP03] [FRS01] These requirements are at the core of the system.
2	[FRP02] [FRP07] [FRA01] Necessary checks to operate smoothly
3	[FRP04] [FRP05] [FRP06]

2.3. Non-Functional Requirements

2.3.1. Compliance with Recognized IoT / Web Standards

"The Internet of Things (IoT) is widely recognised to have lots of potential, but its commercial potential is being held back by fragmentation. A sensor on its own has limited value, but there are huge opportunities for open markets of services that combine sensors, actuators and multiple sources of information. The Web of Things seeks to counter the fragmentation of the IoT, making it much easier to create Applications without the need to master the disparate variety of IoT technologies and standards. Digital twins for sensors, actuators and information services are exposed to consuming Applications as local software objects with properties, actions and events, independently of the physical location of devices or the protocols used to access them." (W3C Web of Things Working Group, 2020)

The Web of Things Interest Group at W3C[1] leads standardization for the so-called Web of Things (WoT). W3C WoT standards have been identified as the most relevant standard series for the smart buildings use cases (see extract from the WoT architecture presented above).

Relevant WoT standards with their current status are reported on Table 4: Status of W3C WoT Standards:

TABLE 4: STATUS OF W3C WOT STANDARDS

Title	Date for current version	URL for the latest published version	Content description	Status
Web of Things (WoT): Use Cases and Requirements	27 January 2021	https://w3c.github.io/wot-usecases/	Collection of new IoT use cases from various domains	Draft
Web of Things (WoT) Architecture	9 April 2020	https://www.w3.org/TR/wot-architecture/	Description of the abstract architecture for the W3C Web of Thing	Recommended
Web of Things (WoT) Thing Description 1.1	27 January 2021	https://www.w3.org/TR/wot-thing-description/	Formal model and a common representation for a Web of Things (WoT) Thing Description	Recommended
Web of Things (WoT) Binding Templates	30 January 2020	https://www.w3.org/TR/wot-binding-templates/	Binding Templates enable a Thing Description to be adapted to the specific protocol or data payload usage across the different standards.	Working Group Note (draft)
Web of Things (WoT) Scripting API	24 November 2020	https://www.w3.org/TR/wot-scripting-api/	Application programming interface (API) representing the WoT	Working Group Note (draft)

[1] https://www.w3.org/WoT/WG/

			Interface that allows scripts to discover, operate Things and to expose locally defined Things.	
Web of Things (WoT) Security and Privacy Guidelines	6 November 2019	https://www.w3.org/TR/wot-security/	Guidance on Web of Things (WoT) security and privacy.	Working Group Note (draft)

Due to the early stage of the standardization of the Web of Things, some W3C standards are still in draft version and can be significantly modified. This fact is accepted by the writers of this document, recognizing that it is more important to adhere to standards, even in evolution, and to adapt the IoT Ecosystem if there are changes.

The IoT Ecosystem for buildings shall specialize the WoT Architecture for the specific context of smart buildings.

Smart System descriptions shall be compliant with the WoT Thing Description (TD) (W3C Web of Things Working Group, 2021). Links to Smart Systems on the field should follow the WoT Binding Templates guidelines (W3C Web of Things Working Group, 2020).

The WoT Scripting API (W3C Web of Things Working Group, 2020) is described in a working document and features a reference implementation[2]. It could be a valuable tool to upgrade the Platforms but is not a compulsory element of the IoT Ecosystem for buildings.

The WoT Security and Privacy Guidelines (W3C Web of things Working Group, 2019) describes threat scenarios and proposes recommendations based on the best available practices in the industry. As it contains informative statements only, it is not part of the IoT Ecosystem.

2.3.2. Requirements on Privacy

A Platform shall be able to grant or deny Applications the right to access single monitoring or control points[3].

Ensuring a legitimate use of the collected data (e.g., purpose limitation and data minimization according to GDPR) is the sole responsibility of the Application.

Platforms play a technical mediation role and do not store any personal data. Hence, they are not subjected to data protection regulations. However, they support Customers for privacy management. As part of the subscription process to an Application, a Platform should give access to monitoring or control parameters only after a formal approval by the Customer.

[2] http://www.thingweb.io/

[3] In the domOS vision, the access rights for applications should be managed by Customers. The implementation of such a privacy management system is not included in the requirements.

Platforms are also able to present to Customers a report on implemented access rights (which Application may access which monitoring / control parameters). It is assumed that when a customer registers on a platform, he grants the platform the rights to control access to the Smart Systems belonging to him and which are also registered on that platform. If the customer no longer wants this control to be carried out, he will have to deregister from the platform.

2.3.3. Security Requirements

Smart Systems are assumed to feature a secure (i.e., authenticated and encrypted) data interface.

Security credentials for Smart Systems shall be uploaded in the Platform, e.g., as part of their description.

Applications establish a secure connection to Platforms, using a (state-of-the-art) security scheme and a security credential provided by Platforms.

Applications shall in no circumstances know Smart Systems security credentials.

2.3.4. Safety

Smart Systems shall work safely even if the connection to the Platform or to an Application is lost. If a Smart System does not meet this requirement, it should not be part of the Ecosystem. Hence, network or Platform failures may decrease the performance of the subscribed Smart Systems but shall not jeopardize their safety.

The Platform operators decline any liability regarding the safety of connected Smart Systems.

An Application could cause safety problems by providing erroneous set points (e.g., frequent switch on / switch off commands on a heat pump). The liability for such safety issues shall be handled directly by Customers and Application operators.

2.3.5. Usability

Management procedures for registering Customers, adding / removing Smart Systems, and subscribing / unsubscribing to Applications shall be simple enough to be executed by ordinary persons.

The usability of management procedures depends on the concrete implementation of a Platform but also on features of the IoT Ecosystem. The IoT Ecosystem should enable the development of compliant Platforms featuring a high degree of usability.

2.3.6. Performance

The IoT Ecosystem should allow scalable implementations. This means, but is not limited to:

- using lightweight protocols,
- enabling implementation in independent stateless modules,
- minimizing the volume of stored data,
- allowing parallel implementation in clusters.

3. Scenarios

3.1. Introduction

The different scenarios presented in this section should help developers to implement their software solution. These scenarios will be used in particular for the validation of the PoC. Four scenarios are represented in a logical order of operations. Each scenario builds on the previous ones:

- the process of subscribing a smart system to a platform
- the maintenance of a directory by the platform
- the process of subscribing a customer to an application
- the functioning of communication between an application and a smart system, highlighting the role of the platform as an intermediary function.

3.2. Smart System Description

The IoT Ecosystem shall provide guidelines on the schema of the Smart System description. These guidelines shall be based on the Things Description specification defined by the WoT architecture.

The Smart System description shall use of the domOS core ontology, to name monitoring and control parameters in participating Smart Systems [FRS01].

Platforms should have access to the description of registered Smart Systems.

3.3. Smart Systems Directory

The scenario “Subscription to an Application” presented in Section 3.4 requires that the Platform manages a Smart System directory. A Smart System directory is a collection of all Smart Systems registered to a particular Platform and differs to the description of Smart Systems.

The description of Smart Systems “Thing Directory” should be uploaded in the directory. During the subscription process, Applications should express their requirements in such a way that the Platform can check whether the requirements are fulfilled by asking the directory. The WoT Architecture defines a “Thing Directory” role and refers to the IETF draft standard (IETF).

3.4. Subscription to an Application

The following prerequisites are assumed to be fulfilled: Customer C is registered on the Platform P, Application A are registered on the Platform P, Smart Systems SS1 and SS2 have registered their description on Platform P.

In this context:

- Customer C initiates a subscription process to Application A.
- Platform P verifies that the Smart Systems SS1 and SS2 comply with A’s requirements. If not, the subscription process is aborted. The verification process makes use of the Directory scenario presented in Section 2.2.7.

- Platform P enables A to access the required parameters after approval of the Customer C. The Application A is from now on active.

This subscription process is also illustrated in Figure 3.

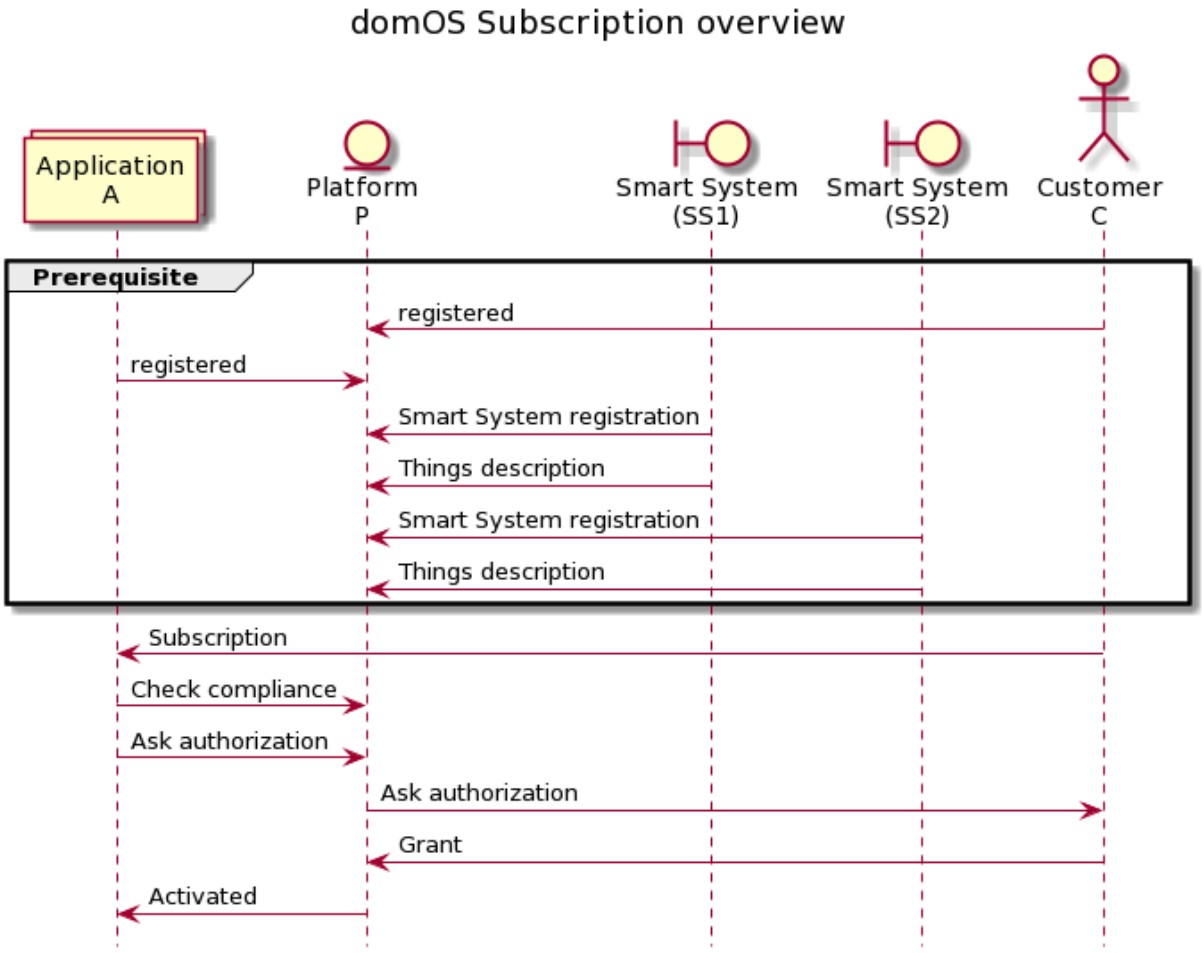


FIGURE 3: ILLUSTRATION OF THE SUBSCRIPTION PROCESS

3.5. Intermediary Function

The intermediary function as introduced in Section 2.2.2 Requirements for Platformsshould be based on the Intermediary concept defined in the WoT architecture[4]. This concept is illustrated in **Error! Reference source not found.**.

[4] "*An entity between Consumers* (Applications in the present document) *and Things* (Smart systems) *that can proxy, augment, or compose Things and republish a WoT Thing Description that points to the WoT Interface on the Intermediary instead of the original Thing. For Consumers, an Intermediary may be indistinguishable from a Thing, following the Layered System constraint of REST.*" Web of Things (WoT) Architecture. W3C Recommendation 9 April 2020

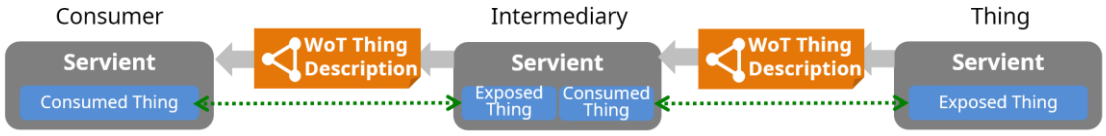


FIGURE 4: ILLUSTRATION OF THE INTERMEDIARY CONCEPT IN THE WOT ARCHITECTURE [WOT ARCHITECTURE] (CONSUMER: APPLICATION IN THE PRESENT DOCUMENT, THING: SMART SYSTEM, THING DESCRIPTION: SMART SYSTEM DESCRIPTION

The intermediary acts as a proxy between Applications and Things.

The implementation of the intermediary should be based on the servient concept defined in [WoT]. As illustrated in **Error! Reference source not found.**, Smart Systems "expose" their descriptions ("Thing Description") to the servient that "consume" them. The servient merges the descriptions into a new description, "exposed" to Applications.

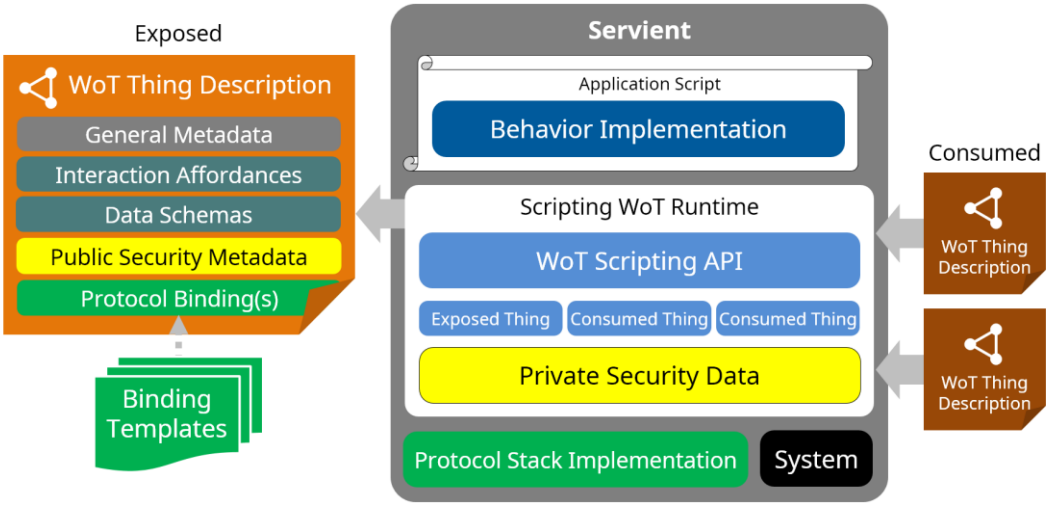


FIGURE 5: IMPLEMENTATION OF THE INTERMEDIARY BASED ON THE SERVIENT CONCEPT [WOT ARCHITECTURE]

The servient implementation should make use of the WoT Scripting API [WoT Scripting API], which allows an abstract[5] and semantic access to Smart Systems (Things). This common semantic (that will be defined in WP3) will allow any application to understand the Thing, allowing, the communication (protocols and message formats) between an Application and the Platform to differ from the communication between the Platform and a Thing.

The intermediary should implement the security as defined in Section 2.3.3.

[5] In this context "abstract" means "independent of communication protocols and message formats".

4. Conclusion

In this document, the basis for setting up an interoperable infrastructure for old buildings has been exposed. This infrastructure introduces the concept of Application, Platform, and Smart Service. The Platform, a pure ICT player, acts as an intermediary between Applications and Smart Systems. One or more applications can provide services to Smart Systems, which are composed of hardware connected to a building.

Functional and Non-Functional requirements such as security, safety, privacy, semantics, usability, performance, and deployment topologies have been exposed. Particular emphasis was placed on compliance with recognized IoT and web standards.

Through scenarios, it is shown how a smart system should reflect its environment and how it connects to the platform. The role of the intermediary function performed by the platform (servient) is also shown.

Retrofitting an old installation and rendering it compatible with domOS, making it accessible for multiple services, should be as simple as connecting sensors, actuators, a communications level, and a provision of a description of the connected system. Further work will focus on the aspects of the actual implementation of this ecosystem.

5. References

IETF CoRE Resource Directory (Draft) [Online]. - https://tools.ietf.org/html/draft-ietf-core-resource-directory-21.

W3C Web of Things Working Group Web of Things (WoT) Architecture [Online]. - 9 April 2020. - https://www.w3.org/TR/wot-architecture/.

W3C Web of Things Working Group Web of Things (WoT) Binding Templates [Online]. - 30 January 2020. - https://www.w3.org/TR/2020/NOTE-wot-binding-templates-20200130/.

W3C Web of Things Working Group Web of Things (WoT) Scripting API [Online]. - 24 November 2020. - https://www.w3.org/TR/wot-scripting-api/.

W3C Web of things Working Group Web of Things (WoT) Security and Privacy Guidelines [Online]. - 6 November 2019. - https://www.w3.org/TR/wot-security/.

W3C Web of Things Working Group Web of Things (WoT) Thing Description 1.1 [Online]. - 27 January 2021. - https://w3c.github.io/wot-thing-description/.