UNIVERSIDAD DE EXTREMADURA

Tesis Doctoral

Situational-Context: un paradigma para automatizar el comportamiento de los dispositivos de la Internet de las Cosas a las preferencias de las personas

Daniel Flores-Martin

Programa de Doctorado en Tecnologías Informáticas (TIN)

Con la conformidad de los directores

Dr. José Javier Berrocal Olmeda y **Dr. Juan Manuel Murillo Rodríguez**

Esta tesis cuenta con la autorización del director/a y coodirector/a de la misma y de la Comisión Académica del programa. Dichas autorizaciones constan en el Servicio de la Escuela Internacional de Doctorado de la Universidad de Extremadura.

2023

A mis padres Antonio y Blanca,
y a mis hermanos Antoñin y Blanqui,
por su paciencia, por comprenderme,
y por apoyarme a lo largo de este camino.

A mi cuñado Antonio,
por descubrirme el mundo de la informática,
y por despertar en mi la ilusión
para dedicarme a ello.

A mis sobrinas Beatriz, Candela, Leyre, y África,
unas de mis mayores fuentes de felicidad y amor,
para su inspiración.

Institutional acknowledgements

I would like to acknowledge the support provided by the following institutions during the completion of this thesis:

- To the Ministry of Science, Innovation and Universities, for supporting me through the grant "Formación del Profesorado Universitario", FPU17/02251.
- To the Ministry of Economy and Competitiveness (MINECO, UE) - European Regional Development Fund (FEDER), for supporting the projects TIN2015-69957-R and RTI2018-094591-B-I00.
- To the Department of Economy, Science and Digital Agenda of the Extremadura Government, for the funding of the project IB18030.
- To the Interreg V-A España-Portugal (POCTEP) 2014-2020 program and the European Regional Development Fund (FEDER) for the financing of projects 0045-4IE-4-P and 0499_4IE_PLUS_4_E.
- To the Regional Government of Extremadura-Ministry of Economy and Infrastructures and the European Regional Development Fund (FEDER), through grants GR15098, GR18112, and GR21133.

Personal acknowledgements

"Forever trusting who we are, and nothing else matters."
Metallica

In 2014, I started my undergraduate studies at the University of Extremadura. For me, it was a new world, something I did not know, and of which I had no background of any kind. They were challenging but very beautiful years, in which I learned many things and met wonderful people. However, I never imagined that my dream of becoming a computer engineer would go even further and lead me to do, first, two master's degrees and then, this doctoral thesis, now in 2023.

I would like to thank my directors Juanma and Javi, for the time invested in me, for having guided me along the way, and even for having acted as psychologists on some occasions. Their good practices and advice have taught me a lot about life as a researcher.

Also, I would like to thank my colleagues in the Spilab laboratory, with whom I have shared so many good moments, both personal and professional, and who in one way or another have helped me to grow: José Manuel, Jaime Galán, Marino, Quique, Cristina, Carlos Canal, Sheila, Borja, Sara Arroyo, Luis López-Lago, Sara Chimento, Isa, Manuel Azabal, Jaime Alvarado, Juanlu, Rubén, Sergio, Javi Romero, and Javi Rojo. Special thanks also to Silvia, for all the support offered for the management of the documentation for trips, conferences, calls, etc. She is an example of hard work, always ready to help us with whatever we need and with a smile on her face no matter what.

I want to thank the Quercus research group in general, and especially Juan, for welcoming me and making me part of the rights and duties that a research group entails. There were many people who helped me when I arrived and even today some of them still care about me.

I would also like to thank my colleagues in the Department of Computer and Telematic Systems Engineering, and especially Pedro, a department that I have been part of since I arrived, for their support, for allowing me to be part of it, and for assuming the appropriate teaching competences that have also taught me how the teaching path within the university works.

Also, I would like to thank all the professors who have passed through my university life and have trained me as an engineer, since I started my degree at the University Center of Mérida until I finished the two masters at the Polytechnic School. I have learned a lot from most of them and I have fond memories of them.

Last but not least, I would like to thank my family and friends for accompanying me during this adventure. An adventure full of ups and downs and where your support has been fundamental in improving me day by day. Support that you have offered me directly or indirectly and even without you realizing it, but for me has been very important and I really appreciate it. Without you, this way would not have been the same. Thank you, Fran, Elena, María, Diego, and Ángela.

Surely I will forget someone to thank because there are many people who have passed through my life during these years. But to all of you, thank you for being there.

Dani

Resumen

La Internet de las Cosas (IoT) está cada vez más presente en nuestro día a día. La IoT es una red de dispositivos con conectividad de red que pueden intercambiar información entre sí. Gracias a la evolución de las tecnologías, el número de estos dispositivos es cada vez mayor. Son muchos los dominios donde se encuentran estos dispositivos, como el hogar inteligente (lámparas, humidificadores, aires acondicionados, etc.), el cuidado de la salud (oxímetros, medidores de ritmo cardiaco, monitorización de la glucosa en sangre, etc.), el transporte (coches inteligentes, semáforos, calculadores de rutas, etc.), la industria (sensores de monitorización, máquinas de proceso en serie, dispositivos de prevención de riesgos laborales, etc.) o la ciudad inteligente (monitorización de la calidad del aire, alumbrado inteligente de calles, tratamiento de la congestión del tráfico, etc.).

Para obtener el mayor rendimiento de los dispositivos, estos deben ser configurados manualmente. Esto puede suponer un problema, primero, debido al gran número de dispositivos que tenemos alrededor y al tiempo que supone configurarlos, y, segundo, a que esta configuración necesita de unos conocimientos mínimos para ser llevada a cabo. El paradigma Situational-Context que se presenta en esta Tesis persigue contrarrestar estos problemas. Por lo tanto, en esta tesis se abordan los siguientes desafíos:

- Favorecer la creación de entornos heterogéneos, permitiendo la integración de dispositivos independientemente del fabricante y el protocolo de comunicación.

- Describir de manera unificada la información de las personas indicando sus preferencias, y que será utilizada para adaptar el comportamiento de los dispositivos al entorno.

- Identificar en tiempo real y a partir de estas preferencias las necesidades de las personas y localizar algún dispositivo del entorno que pueda solventarla.

- Modificar automáticamente el estado del entorno adaptando el comportamiento de los dispositivos a las necesidades detectadas.

Para abordar estos desafíos se ha desarrollado una arquitectura que permite conectar personas y dispositivos independientemente del fabricante, y que es capaz de reconocer las preferencias de las personas para adaptar el entorno. Finalmente, esta arquitectura ha sido validada a través del desarrollo de un caso de estudio basado en una oficina inteligente, en el que se ha comprobado la viabilidad y eficacia.

Con el desarrollo del Situational-Context se consigue adaptar el comportamiento de los dispositivos IoT a las necesidades de las personas de forma automática, sencilla, colaborativa y fácilmente aplicable a cualquier dominio.

Abstract

The Internet of Things (IoT) is increasingly present in our daily lives. The IoT is a network of devices with network connectivity that can exchange information with each other. Thanks to the evolution of technologies, the number of these devices is increasing. There are many domains where these devices can be found, such as the smart home (lamps, humidifiers, air conditioners, etc.), healthcare (oximeters, heart rate meters, blood glucose monitoring, etc.), transportation (smart cars, traffic lights, route calculators, etc.), industry (monitoring sensors, serial processing machines, occupational risk prevention devices, etc.) or smart cities (air quality monitoring, smart street lighting, traffic congestion management, etc.).

To obtain the maximum benefit from these devices, they must be configured manually. However, this can be a problem. First, because of the large number of devices around us and the time, it would take to configure them, and second, because this configuration requires minimal technical knowledge. The Situational-Context paradigm presented in this thesis seeks to counteract these two problems. Therefore, the following challenges are addressed in this Thesis:

- To favor the creation of heterogeneous environments, allowing the integration of devices regardless of the manufacturer and the communication protocol.
- To describe in a unified way the information of people indicating their preferences, which will be used to adapt the behavior of the devices to the environment.
- To identify in real-time and from these preferences the needs of people and locate any device in the environment that can solve them.

- To modify automatically the state of the environment adapting the behavior of the devices to the detected needs.

To address these challenges, an architecture has been developed that facilitates the connection between people and devices regardless of manufacturer. Also, the architecture offers support to recognize people's preferences to adapt to the environment. Finally, this architecture has been validated by developing a case study based on a smart office, in which the feasibility and effectiveness have been tested.

With Situational-Context, the behavior of IoT devices can be adapted to people's preferences automatically and without user intervention. Furthermore, this solution can be easily applied to any IoT domain, as it facilitates the creation of heterogeneous environments regardless of manufacturers and communication protocols. This allows users to create their smart environments affordably and with all types of devices.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

"Somewhere, something incredible is waiting to be known."
Carl Sagan

Contents

1.1 Thesis Origins

The Quercus Software Engineering [1] group addresses different lines of research among which is social and pervasive computing, whose responsibility

[1]https://quercusseg.unex.es

falls on the Spilab [2] (Social and Pervasive Innovation Lab) laboratory. This line focuses on the development of applications where social interactions and people are the protagonists. Interactions produce a large amount of data that can be used for different purposes, such as statistical studies, personalizing services, or offering customized content. These interactions are also conditioned in many cases by different contextual elements such as spatial-temporal conditions, weather, other people nearby, smart devices, or any other element that may influence them. In this sense, within the Spilab work team was born the concern to combine the interactions of people with the context in which they are. The first works were focused on proposing the PeaaS (People as a Service) paradigm, a model for offering information about people as services to create more personalized applications. This information is variable depending on where it is applied but can be location, personal preferences, or even interactions with other people. From this idea, the IoP (Internet of People) was born, where due to the great importance that people were gaining in smart environments, the idea of considering people as a central element in the development of applications was arising. The IoP considers people as the main elements of intelligent environments and everything revolves around them. At this point, smart devices are the ones that must modify their behavior to meet people's preferences. Thus, the idea of social environments formed by people and devices grew more and more. From this was born the idea of the Situational-Context, as a paradigm to adapt intelligent environments to the situations where people are involved, taking into account the conditions of the context.

Situational-Context was defined as a theoretical framework in (Berrocal, Garcia-Alonso, Canal, & Murillo, 2016), where the fundamental elements of people, intelligent devices, and context information were identified. In addition, some theoretical examples of the functioning of the paradigm were proposed as a proof of concept to detect the need for its use. These examples allowed, first, to detect what information was necessary to obtain from people and smart devices to be used in the adaptation of the environments, second, what contextual information is relevant to perform a personalized adaptation, and, third, to define an action strategy to carry

[2] https://spilab.es

out this adaptation. A simple example is that of adapting the temperature of a room taking into account the preferences of each person. For this, it is necessary to obtain information about each person, what temperature he/she prefers, what air conditioning devices exist in the room, where they are located, and how they should be used. Analyzing all this, we came to the problem that intelligent environments are very heterogeneous and that the information obtained can come in different formats or through different protocols. This heterogeneity limited, in many cases, the intelligent environments to be created, where only certain devices or certain functionalities were available to ensure compatibility between them. As a consequence, the environments were subject to a specific vendor to ensure compatibility and achieve a more complete adaptation. In environments with devices from different manufacturers, this compatibility was more complex or even nonexistent, and therefore adaptation was very limited. On the other hand, we identified the problem of performing this adaptation manually. Traditionally, people are in charge of manipulating the devices to obtain the desired performance. Continuing with the example, it would be to manipulate the air conditioner to adapt to the temperature of the room. This poses several problems. First, there is the time involved in configuring the devices to achieve the desired performance. Second, the need to have a minimum knowledge of the operation of the device to use it. Thirdly, the operation of the device does not always have to be the same to adapt to the environment, it will depend on the environmental conditions, people involved, etc. And fourth, the greatest complexity arises when there are several devices that must adapt to several people, for example, an air conditioner and a humidifier to regulate the temperature and adapt to the preferences of several people. In view of all this, the creation of collaborative and proactive environments arises.

Many solutions address these problems in one way or another. However, these solutions focus on achieving more effective adaptation to environments and not that it happens automatically or by taking people's information into account. Although the context is considered in many current solutions, it is not taken into account in terms of people's information, which is continuously changing and on which the operation of the devices must depend. Therefore, we find that these problems are not addressed jointly in the current works, which led the Spilab team to out-line the Situational-Context and which has motivated its full development through this doctoral thesis. In this thesis, we address the problem of creating smart environments, paying particular attention to the heterogeneity of devices, the collaboration among devices and people, and the automatic adaptation to the environment.

1.2 Research Context

Since the arrival of the Internet into our lives, it has evolved rapidly and overwhelmingly, from those rudimentary 56 Kbps modems to today's very fast and efficient fiber optic lines. The evolution of the Internet has been accompanied by the development of devices that, thanks to this connection, can consult millions of databases and exchange information. Today, we can connect to the Internet on our mobile phones, printers, smart TVs, IP cameras, GPS and a multitude of electronic devices that have this interface, but: **have we reached the limit or is it just the beginning of a new era of the Internet?**

In recent years, we began to talk about the Internet of Things (IoT) as a promising bet for the future and which nowadays is beginning to become an absolute reality (Li, Xu, & Zhao, 2015). The IoT is made up of millions of smart devices connected to the Internet. The purpose of these devices is to make people's lives easier, simplifying tasks or making them automatic.

The evolution of the IoT allows developers to design increasingly smart devices and truly useful services that a few years ago were unthinkable, such as lighting or temperature controller, devices checking blood pressure, or detecting and notifying emergency situations. Thus, companies can offer services linked to a product or a specific product, so that the value of the business can be derived from the exploitation of these services, thus achieving an additional income to the sale of the product in question. These services make it possible to consult information or perform actions of practically any type and from any device.

However, to get the maximum benefit from them, they must collaborate with each other to perform complex tasks (Atzori, Iera, & Morabito, 2010). This collaboration will make it possible to offer more personalized and powerful services by combining several services to perform a given action. So, the next evolution of the IoT is to ensure that smart devices can proactively collaborate (Yafei, Guanyu, & Hui, 2016; Al-Fuqaha, Guizani, Mohammadi, Aledhari, & Ayyash, 2015a), as it is also pursued in the programmable world (Taivalsaari & Mikkonen, 2017).

Unfortunately, the possibility of collaboration among smart devices is still far from being achieved. Indeed, manufacturers develop their own protocols so that their devices are endogamic from a collaboration point of view, meaning that they can interact but can not be integrated with devices from other manufacturers. This allows manufacturers to save their market share. Therefore, this practice not only limits the ability of devices from different manufacturers to collaborate but also inevitably leads to the well-known *vendor lock-in* problem (Roman, Zhou, & Lopez, 2013). This phenomenon implies that if one wants to obtain the maximum benefit from the IoT, (s)he must purchase devices from the same manufacturer to ensure maximum compatibility. Consequently, the user experience is limited to the interests of manufacturers.

The collaboration should not be random or fortuitous, it should be driven by the context of the device and the users. For example, in a room with elevated temperature, a humidifier and an air conditioner could modify their state through the invocation of services to regulate the temperature. In this way, the preferences of the people in the room would be solved. The context in which the devices are located is important and provides valuable information about other devices, people's data, weather factors, date and time, etc. Continuing with the example, people's temperature preference will vary depending on whether they are at home, at work, shopping, etc., or whether they are with friends, or alone. Therefore, the behavior of devices must take into account the characteristics of the context if we want to achieve the most accurate adaptation to people's needs.

The development of context-sensitive software has been successful (Perera, Zaslavsky, Christen, & Georgakopoulos, 2014). IoT devices are becoming more and more intelligent thanks to the information gathered from the context in which they are located. A still very present problem is that the interaction of people with IoT devices is still too manual (Shrestha, Kubler, & Främling, 2014; J. Kim et al., 2016), who often do not have the necessary technical skills, with the consequent investment of time and frustration that this can cause. Nowadays, when we buy a new device, we have to spend some time configuring it to behave the way we want it to.

In addition to this, in many cases, we must have a minimum of knowledge to perform this configuration properly. Continuing with the previous example, if a person acquires a new air conditioner, you have to spend time configuring the hours in which you want it to work, or that only turns on when the ambient temperature is higher than desired, etc., and all this, following the manufacturer's instructions. In this sense, to minimize the interaction of people with the devices, this communication must arise from the context. That the air conditioner learns when to operate, or under what specific circumstances, without the person, must configure anything.

These drawbacks can be addressed by developing software capable of adapting its behavior to the people's needs (Perera et al., 2014; Taivalsaari & Mikkonen, 2017). In addition, several research areas can contribute to solving them, namely Context-Oriented Programming (COP), Ambient Intelligent (AmI), Semantic Web (SW) and Machine Learning (ML). Each of these paradigms can make interesting contributions. COP allows the development of applications whose operation depends on contextual conditions and therefore allows dynamic behavior. AmI pursues the creation of intelligent environments that are sensitive to people, and therefore the behavior of the environments revolves around them. SW makes it possible to describe the information of the environments in a more or less standard way and also to establish a series of relationships between devices and people, facilitating their collaboration. And finally, ML is a paradigm that has been on the rise in recent years and allows devices and applications to be able to learn or predict certain behaviors. Most of these paradigms allow us to define behaviors for different scenarios at the time the systems are designed, so the adaptation of the devices is limited to situations that developers have been able to identify, making it impossible to adapt to other situations that may arise from the context.

The aim of this thesis is to develop a methodology to achieve the collaboration of smart devices regardless of the manufacturer by using semantic web techniques, to satisfy people's needs according to the context they are in. The main objective of the semantic web is to improve the Internet by extending interoperability between computer systems using smart agents and applications that seek information without human intervention (Barnaghi, Wang, Henson, & Taylor, 2012b). The semantic web is a widely used resource to achieve semantic interoperability between services and devices.

We achieve this interoperability by providing smart devices with *goals* and *skills*. These goals and skills are defined in semantic web terms and related by semantic reasoners and query languages. In an environment, nearby devices are related in what we call a *situation*. A situation is defined by the information derived from the context such as the present devices or people, the date, or the location. The parameters that influence a situation are very numerous, and therefore, the situations that can occur are innumerable. Thanks to relationships between goals and skills, different strategies can emerge to solve the detected goals with the available skills, creating a collaborative environment. Therefore, the possibility of collaboration between devices is allowed while maintaining the independence of the manufacturer, without forcing any device or manufacturer to use any specific technology, in a simple way, at a low cost and effectiveness. The feasibility of this proposal is evidenced by different case studies is detailed in the next sections.

Therefore, in this thesis, we develop the paradigm Situational-Context, in which the main objective is to achieve the highest level of comfort in a given situation by adapting smart devices' behavior to people's preferences automatically.

1.3 Problem Statement

IoT is a novel paradigm that is rapidly gaining ground in people's lives. The basic idea of this concept is the widespread presence that surrounds us of a variety of internet-connected things, such as RFID tags, sensors, actuators, mobile phones, etc., which, through unique addressing schemes, are capable of interacting with each other and cooperating with their neighbors to achieve common goals. (Atzori et al., 2010). IoT allows physical devices to see, hear, think and work by making them collaborate, share information and take coordinated decisions.

A growing number of physical objects are being connected to the Internet at an unprecedented rate realizing the idea of the IoT, as we can see in Figure 1.1. Recent estimates state that in the next few years we will have about 25.4 billion smart devices connected to the Internet (Howarth, 2022). IoT applications are seen in several domains and this reflects the significance of IoT. These domains include multiple types of devices dedicated to transportation, healthcare, industrial automation, and emergency response to natural or man-made disasters where human decision-making is difficult, among others (Al-Fuqaha et al., 2015a).

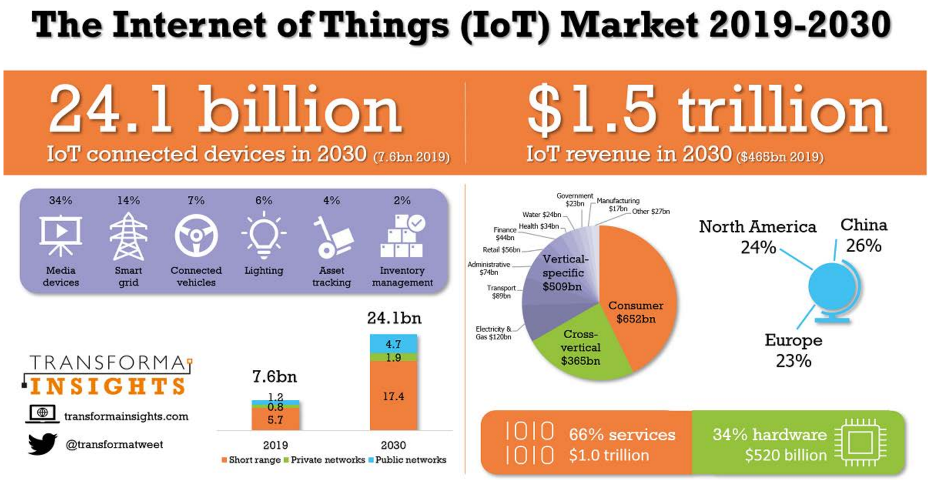


Figure 1.1: The IoT Market 2019-2030 (Insights, 2020)

However, the fact remains that there are many challenges and issues related to the use of IoT, and they cannot be ignored (Zhou, Cao, Dong, & Vasilakos, 2017; Chiang & Zhang, 2016; Van Kranenburg & Bassi, 2012). Some of these challenges are:

1. **Big scale:** The price and optimization can only come by the large scale, in the same way, that the opportunities for growth of the medium and small business are severely curtailed with pilots, tests, and trials. A closed circle that we have to try to break on either side.

2. **Cybersecurity:** Using traditional technologies lightly can bring these risks. The new native IoT technologies have a much safer approach, but we do not know what threats we will encounter in the coming years and we should be preparing for them now and planning mitigation for possible future attacks.

3. **Unified technologies:** There are several technologies to develop IoT

applications. This takes us to a heterogeneous environment where communication can be difficult. We must be able to respond to connectivity in a cheap and universal way in the IoT.

4. **Business intelligence oriented to services development:** We must create services, but also the tools on which we will create them must understand the business and apply Big Data and machine learning to give added value, until now non-existent. Services, not devices, products, not platforms.

Apart from these challenges, the IoT has, as every technology, a number of advantages and disadvantages (Saxena, 2016; RedAlkemi, 2018). Talking about the advantages that the IoT brings to our lives, the most important ones are:

- **Access information:** easy access to data and information that is sitting far from the location. Access to data must be guaranteed to provide appropriate and personalized services. Due to the constant changes in intelligent environments, this access must be in real-time.
- **Communication:** better communication is possible over a network of interconnected devices, making the communication of devices more transparent, which reduces inefficiencies. Processes, where machines have to communicate with each other, are made more efficient and produce better, faster results.
- **Cost-effective:** as mentioned in the point above, communication between electronic devices is made easier because of IoT. This helps people in daily tasks. Transferring data packets over a connected network save time and money.
- **Automation:** automation is the need of the hour to manage everyday tasks without human intervention. Automating tasks in a business helps boost the quality of services and reduce the level of human intervention.

Unfortunately, the IoT does not just bring good things into our lives, there are also some disadvantages that we have to consider:

- **Compatibility:** as of now, there is no standard for tagging and monitoring with sensors. A uniform concept like the USB or Bluetooth is required which should not be that difficult to do.

- **Complexity:** a diverse network that connects various devices is what we call IoT. A single loophole can affect the entire system. This is by far the most complicated aspect of the IoT that can have a tremendous effect.

- **Lesser jobs:** with every task being automated, the need for human labor will reduce drastically. This will have a direct impact on employability.

- **Dependability.** we may not notice it, but we are witnessing a major shift in technology and its implementation in everyday life. There is no doubt that technology is dominating our lifestyle, reflecting a human's dependability on technology.

The Internet of Things allows us to do a lot of interesting things, thanks to the potential of the smart devices we have today, as all its advantages show. However, as mentioned above, there are many drawbacks and challenges as it is still a new technology and needs to be further investigated to take advantage of its full potential. This is a major problem for the evolution of IoT and may condition its development.

For this reason, the aim of this thesis is to solve, as far as possible, the disadvantages shown above. Specifically, by favoring compatibility between different types of devices, ensuring that they can easily exchange information and that the environments make use of them to modify their state automatically. By addressing these challenges, we can take a further step towards a more autonomous, people-centric IoT, which will offer more powerful possibilities and even require less attention from people.

1.4 Thesis Objectives

The main hypothesis of this thesis is the following: **To achieve a dynamic interaction of IoT devices, where the behavior and strategies to**

follow will emerge from the context stored in smartphones, orienting their operation to people's needs.

In addition, it is intended to achieve better integration of people with the IoT by making devices learn about their owners. This thesis has the following specific objectives:

1. **To analyze the conditions necessary to achieve a run-time collaboration in IoT environments**. This analysis must enable the integration of devices with different characteristics to create heterogeneous environments, to subsequently identify the levels of collaboration among devices and people in IoT environments, from initial connectivity, through information exchange, to situation identification.

2. **To establish a data model to define interactions**. This model must allow both people and devices to be able to record a history of interactions with other entities in the environment to identify situations and take actions to adapt to the environment.

3. **To design an architecture where people and devices are represented**. The architecture must support the identification of people and devices present in a situation, their connections, and the automatic management of device behavior according to people's preferences.

4. **To propose a model for learning about people's preferences**. This model must be hosted on smartphones and, therefore, must comply with the resource constraints of these devices. It will be provided as a service from the smartphones to the architecture to learn the preferences of all the people involved.

5. **To evaluate the developed system**. This evaluation must ensure that each of the developed aspects fulfills its function and that the system meets the challenges addressed. In addition, efficiency and performance tests should be performed to detect technical requirements and limitations.

1.5 Thesis Contributions

The objectives stated for this thesis were addressed throughout the design and implementation of the Situational-Context. As a result, this thesis provides a series of contributions to analyze, model, and adapt IoT environments. The main contributions derived from this thesis are detailed below:

1. An **analysis** that lies in the inclusion of a new level of collaboration that allows the detection and creation of situations in real-time from the contextual information of these environments.
2. The **definition of a model** allows describing the necessary properties of smart devices and people to favor their integration and relationship in smart ecosystems.
3. The **develop of an infrastructure** that can contribute to the definition of a standard for connecting people to devices regardless of the manufacturer.
4. The **adaptation of machine learning algorithms** to be executed on smartphones and prepared to handle a significant and larger volume of data.

1.6 Thesis Impact

This section details the impact of the thesis concerning different research projects in which it has been involved, the scientific publications it has generated, and the collaborations established with other universities and researchers.

1.6.1 Research Projects

The different research projects in which the author of this thesis has been involved during the development of this thesis are the following:

- **People as a Service: Habilitando los dispositivos móviles como proveedores de servicio en sistemas ciber-físicos (TIN**

2015-69957-R). A national project whose objective was to enable people to be able to offer and consume services through their mobile devices and that this was not only limited to smart devices. The contribution of this thesis to the project was to use the PeaaS paradigm to define how people can offer services in addition to consuming them. In this way, it was possible to validate PeaaS and prove that it was viable in real case studies.

- **Instituto Internacional de Investigación e Innovación del Envejecimiento (4IE) (0499_4IE_PLUS_4_E)**. Cross-border project (Spain-Portugal) aimed at bringing technology closer to the elderly to improve their quality of life and make their daily lives easier, especially in rural areas where depopulation is a real problem. Among others, the main contribution of this thesis to the project has been the development of a case study based on a nursing home where the environment is adapted to the needs of the elderly through the installed devices. This thesis contributed with numerous papers in workshops where the Situational-Context is integrated into rural areas.

- **Contexto-Situacional: Una arquitectura de gestión de la información personal para una mejor integración personatecnología (RTI2018-094591-BI00)**. A national project where the foundations of the Situational-Context were laid, starting with the requirements of the architecture to its subsequent design and implementation in a real case study. This project is closely related to this thesis. The main contribution has been to develop the architecture for the Situational-Context, defining a common description for devices and people in order to achieve proactive collaboration.

- **Contexto-Situacional: Arquitectura tecnológica para automatizar la conexión de las personas a los dispositivos inteligentes (IB18030)**. Regional scope project whose objective was to enable the interconnection between smart devices and people to favor the automation of tasks in IoT smart environments. This project is also closely related to this thesis. The contribution of this thesis was to define the necessary components for the architecture and to establish the information flow between them to solve the detected problems.

- **Subcontrata de la Universidad de Extremadura por la empresa GAMMA SOLUTIONS, S.L.U. para el desarrollo del caso de uso "Formación y Experiencia Turística mediante realidad mixta" dentro del proyecto "Piloto 5G- Cáceres"**. Scope project Article 83. whose objective is the implementation of 5G technology in the city of Cáceres to improve the connection of smart devices to the Internet. The contribution of this thesis in the project was to improve the integration of devices through different communication protocols, including 5G.

- **QSALUD – Ingeniería del software para computación cuántica aplicada al envejecimiento y la farmacogenética**. Scope project Article 83. which aims to improve the lives of the elderly through the study of the relationship between several variables such as the genetic profile of the individual or their medical history through quantum services that allow predicting how a person will respond to a particular drug. For this project, the main contribution was to leverage the description defined for individuals to store their medical history, among other data.

1.6.2 Summary of Publications

Table 1.1 shows a summary of the forums in which the main papers of this thesis have been published, the scope, the number of publications in each forum, and their importance. The details on these publications can be found in Section 6.2. The importance of conferences (CLASS) is obtained from the GII-GRIN-SCIE (GGS) Conference Rating [3], and the importance of journals is obtained from the Journal Citation Report (JCR) [4].

As can be seen in Table 1.1, a total of 22 papers have been published, of which 7 are national and 15 are international. 18 of these papers were published in congresses/workshops, of which 3 were accepted in congresses indexed in the CLASS ranking. The other four papers were published in

[3] https://scie.lcc.uma.es:8443/gii-grin-scie-rating

[4] https://jcr.clarivate.com

JCR indexed journals, with an impact factor of 4.231 (Q1), 2.420 (Q2), 1.819 (Q3), and 0.617 (Q4).

Table 1.1: Summary of main publications

Forum	**Kind**	**Scope**	**Num**	**CLASS**	**JCR**
International Conference on Service-Oriented Computing (ICSOC)	Conference	Int	1	-	-
International Workshop on Gerontechnology (IWoG)	Workshop	Int	4	-	-
International Conference on Pervasive Computing and Communications (PerCom)	Workshop	Int	2	-	-
International Conference on Web Engineering (ICWE)	Conference	Int	2	3	-
Jornadas de Ingeniería del Software y Bases de Datos (JISBD)	Conference	Nat	2	-	-
IEEE Internet Computing	Journal	Int	1	-	4.231 (Q1)
Future of Information and Communication Conference (FICC)	Conference	Int	1	-	-
Wireless Communications and Mobile Computing	Journal	Int	1	-	1.819 (Q3)
Journal of Web Engineering (JWE)	Journal	Int	1	-	0.617 (Q4)
Symposium on Computers and Communication (ISCC)	Conference	Int	1	3	-
Jornadas de Ciencia e Ingeniería de Servicios (JCIS)	Conference	Nat	5	-	-
Computing	Journal	Int	1	-	2.420 (Q2)
Total papers	**12 Conf.** **6 Works.** **4 Jour.**	**15 Int.** **7 Nat.**	**22**	**3**	**4**

1.6.3 Collaborations

During the development of this thesis, both national and international relationships have been established. We have worked jointly with Professor Carlos Canal, from the University of Málaga (Spain), with Professors Tommi Mikkonen and Niko Mäkitalo from the University of Helsinki (Finland), with Professor Luca Foschini, from the University of Bologna (Italy), and with Professor César Fonseca from the University of Évora (Portugal).

As a result of these collaborations, Professor Carlos Canal's experience in the field of service-oriented computing provided me with a solid basis for defining the basic services to be provided by the Situational-Context. This was because part of his research was focused on using Digital Avatars to compose people-oriented services. Professor Tommi Mikkonen and Dr. Niko Mäkitalo developed a human data model that was adapted to the architecture defined for Situational-Context and thanks to which people's preferences can be defined. In addition, the experience of Dr. Luca Foschini in the treatment of contextual data and recommender systems favored the development of a model for the intelligent selection of services in IoT environments. Finally, Professor César Fonseca's experience in the healthcare field allowed the application of Situational-Context in IoT environments focused on elderly people and facilitating their daily tasks.

Finally, two international stays were conducted, the first one at the University of Helsinki with a duration of 2 months under the supervision of Professor Tommi Mikkonen, and the other one at the University of Évora with a duration of 2 months under the supervision of Professor César Fonseca.

1.7 Thesis Structure

The rest of the thesis is structured as follows:

- **Chapter 1: Introduction**. In this chapter, the work to be done in this thesis has been put into context and some fundamental concepts have been described. It has also shown the main motivations that led to its development, as well as the objectives set to solve the problems detected. Finally, the impact of this thesis has been detailed, both at the level of research projects in which it has been involved, as well as in collaborations with other universities and researchers, and at a personal level in terms of scientific publications.

- **Chapter 2: State of the Art**. This chapter delves into the current state of the IoT by reviewing the most relevant technologies for intelligent environment adaptation and context identification. In addition, a detailed literature review is provided on the main problem to be covered in this thesis: the interoperability and collaboration of IoT devices. This review details the objectives pursued in this review, the research method followed, the search criteria used, the classification taxonomy performed and a summary of the results obtained. All this is intended to further motivate the detected problem and to discover the gap where this thesis can make a real contribution.

- **Chapter 3: Situational-Context: Introduction and Developed Architecture**. This chapter details the concept of Situational-Context and the contributions it can have on IoT environments. It also details the proposed architecture to support the interoperability and collaboration of devices and people in intelligent environments. For this architecture, all its components are explained and a real implementation is conducted.

- **Chapter 4: Situational-Context: Learning Model**. This chapter takes a step further on the proposed architecture and realizes a learning model for people's routines and habits and their interaction with smart devices. For this purpose, several frameworks applied to the Situational-Context are described and the basis for designing applications that allow people to be both consumers and providers of services in intelligent environments are detailed.

- **Chapter 5: Verification and Validation**. This chapter validates the architecture and learning model developed. For this purpose, two case studies are conducted on which the Situational-Context is applied and a series of tests are performed to check the viability and efficiency of the system. In a complementary way, simulations for larger environments are developed to check the scalability of the system as well as some of its limitations.

- **Chapter 6: Conclusions and Future Work**. Finally, this chapter details the conclusions and future work of this thesis. In addition, there is a discussion of the work conducted with the contributions and limitations detected, and a personal reflection on the work as a whole. Finally, the publications derived from the thesis are detailed.

Chapter 2

State of the Art

"El que lee mucho y anda mucho, ve mucho y sabe mucho."
Miguel de Cervantes

Contents

2.1 Introduction

The development of the Situational-Context allows collaboration among people and devices to promote social environments requires analyzing different aspects. First, it is necessary to review the current technologies that enable the development of applications according to the characteristics of the context in which they are deployed. Second, it is also necessary to analyze the possibilities that exist for integrating people into smart environments so that, in addition to consuming services, they can provide them. In this way, a fully collaborative environment is achieved where both smart devices and people can consume and provide services. Third, mechanisms are also needed to specify information from devices and people in a unified way. This is particularly delicate due to the enormous number of different formats that can be used, so it is necessary to explore the currently available paths and assess which is the most appropriate. This information can be stored and processed in different ways, so it is also necessary to analyze which methodology is the most appropriate for our development.

In this section, we review the areas related to the objectives of this thesis to identify different situations in intelligent environments. These areas are related to the analysis of the environmental conditions to deploy ambient intelligence. In addition, these areas provide the necessary background to establish mechanisms that favor the identification of situations and then proactively promote collaboration among devices. Once the different areas are analyzed, a review of the most relevant works in them is conducted, to detect the gap where this thesis finds its contribution.

Thus, in this chapter, we first review the interoperability levels necessary to get collaboration among devices in Section 2.2. Secondly, review the areas related in Section 2.3. And thirdly, once these aspects have been reviewed, Section 2.4 details the systematic review of the literature carried out in this thesis, which encompasses, in addition to the three previous aspects, the characteristics necessary to manage situations dynamically.

2.2 The Collaboration in IoT Environments

Different levels of interoperability are suggested by the current literature to obtain an ideal IoT ecosystem. Achieving these levels will ensure full collaboration between devices in heterogeneous environments. Some of these levels are (Patel, Patel, et al., 2016): *technical*, *syntactical*, *semantical* and *organisational*; (Noura, Atiquzzaman, & Gaedke, 2019a): *devices*, *network*, *syntactical*, *semantic* and *platform*; (Miori, Russo, & Ferrucci, 2019): *basic connectivity*, *network* and *syntactic*; (Elkhodr, Shahrestani, & Cheung, 2016): *technical*, *semantic*, *syntactic*, and *cross-domain*. These levels generally overlap although with different nomenclatures.

In this sense, we proposed in (Flores-Martin, Berrocal, García-Alonso, & Murillo, 2022) the levels that should be addressed to obtain full interoperability, and also provide support for the situations that occur in the environments. These levels are illustrated in Figure 2.1 and are described as follows:

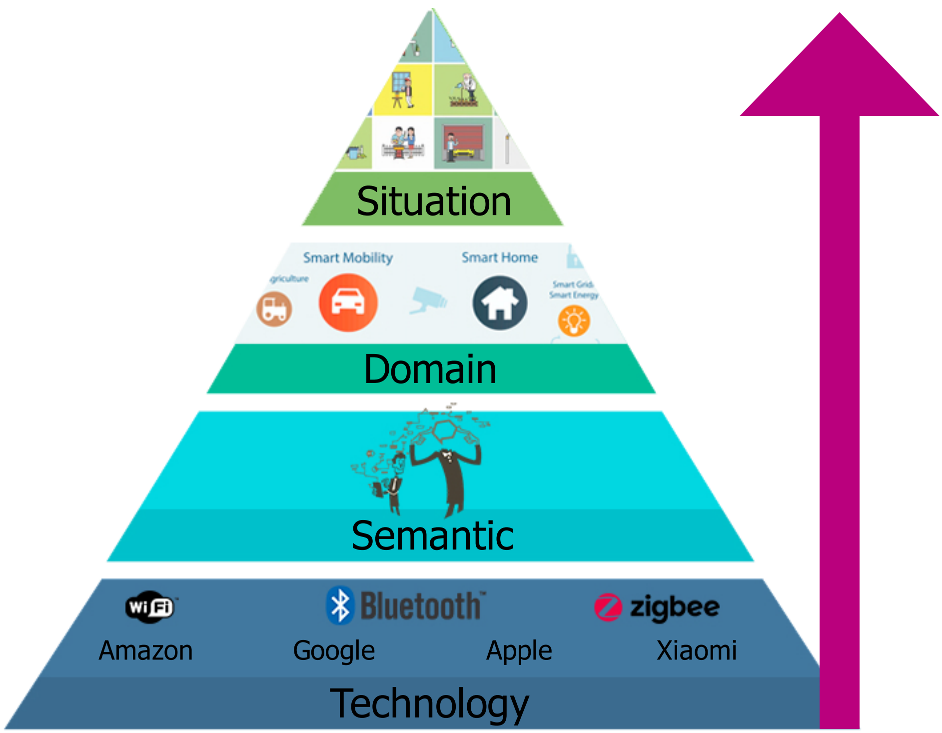


Figure 2.1: Interoperability levels

1. **Technology:** the diversity of manufacturers makes communication between devices from different manufacturers difficult because each manufacturer develops its own communication mechanisms, protocols, and technologies. This layer improves interoperability through communication protocols.

2. **Semantic:** once the communication is established, it is necessary to know the semantic description of the devices. This must specify in a clear and understandable way what information, services, or parameters they have. Smart devices can have similar characteristics or provide similar services. However, we still must get all their details to know how to interact with them.

3. **Domain:** devices designed for a specific domain should be reused to perform other complementary tasks and to interact with devices from different IoT domains. To improve this interaction, the benefits of the semantic layer are considered, such as the definition of device schemes or the use of query or reasoning languages to establish relationships.

4. **Situation:** detecting a particular situation of the environment and its characteristics are key to detailing how devices should collaborate to achieve environmental goals. Even when we know the services provided by a specific device, its domain, and its semantics, each situation requires these services to work in a specific way. Hence, it is necessary to be aware of the context and its attributes that define different situations. These factors can be people in the environment, IoT devices, when and where the situation occurs, or what objectives are being pursued.

The first three levels of interoperability have been widely addressed by the scientific community. In *technological* interoperability, solutions such as (Morabito, Petrolo, Loscri, & Mitton, 2018; Yacchirema, Palau, & Esteve, 2017) are developed to allow devices to be interconnected. These solutions are usually dedicated gateways to facilitate the connection through the invocation of services or microservices. There are also standards defined, for instance, by the IEEE (IEEE, 2022) which are being introduced to facilitate the connection and exchange of data among devices. Also, *semantic* interoperability is being increasingly addressed in (Kaebisch, Kamiya, McCool,

& Charpenay, 2019; Maarala, Su, & Riekki, 2017) through technologies such as the Semantic Web and the use of ontologies and semantic reasoners. As for the *domain* interoperability, there have been some works to allow devices belonging to different domains to connect with each other (J. Kim et al., 2016; Gyrard, Datta, Bonnet, & Boudaoud, 2015a). This is achieved through a complete description of the devices and the use of techniques such as the Semantic Web to establish a coherent relationship between them.

While these levels improve IoT interoperability, the management of different situations has not yet been fully addressed. This is why the Situational-Context has been developed with these levels of interoperability in mind to provide the highest possible interoperability in IoT environments.

2.3 The Pillars

The main pillar on which this thesis revolves is to achieve the adaptation of intelligent environments based on people's information and with minimum interaction with the devices depending on the context. To achieve this, we based on the identification of situations that contain the necessary information to adapt the behavior of the devices. There are different dimensions that make an important contribution to the development of Situational-Context. These dimensions range from the development of lower-level applications to the selection of the necessary services to adapt to the environment, including the necessary mechanisms to favor communication and interaction among devices and people. Within these dimensions, we highlight the most representative aspects that we believe can help with the development of Situational-Context. We find COP as a technique for the development of context-oriented applications, which means obtaining the necessary knowledge to know how these applications behave and how they should be developed. We also have the Semantic Web, which allows identifying in a more or less standard way the devices in smart environments to establish communication and promote collaboration. Interaction based on people is studied through the People as a Service paradigm, which makes it possible to send and receive information from people through personalized services. And finally, the management of intelligent environments is analyzed through the Ambient Intelligence paradigm, which provides us with a global vision of the management of people and devices that must be done in intelligent environments.

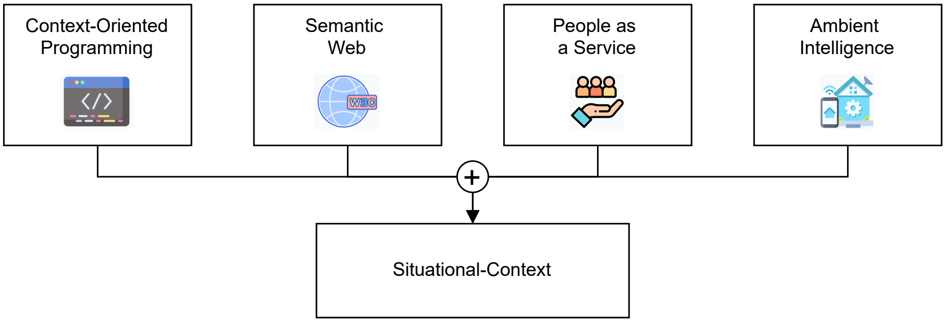


Figure 2.2: Related areas

We consider these four areas as the pillars of the development of this thesis (Figure 2.2). Therefore, these concepts, Context-Oriented Programming, Semantic Web, People as a Service, and Ambient Intelligence, will be introduced below in Section 2.3.1, Section 2.3.2, Section 2.3.3, and Section 2.3.4, respectively. With these concepts, the reader is provided with the necessary background on the importance of context in intelligent environments.

2.3.1 Context-Oriented Programming

The automatic adaptation of applications depends on the context in which they are located. To do this, it is necessary to interpret what is happening in the environment to adapt its operation. COP is a programming paradigm that allows the dynamic adaptation of software according to the execution context. Context is defined as any computationally accessible information. However, most of the time context is considered as information detected from the environment by sensors (location, brightness, temperature, etc.), actuators (light bulbs, air conditioner, plugs, etc.), or internal system changes (resource consumption, energy consumption, bandwidth, etc.).

One of the most important and pioneering works is the one developed by Hirschfeld et al. in (Hirschfeld, Costanza, & Nierstrasz, 2008). It states that one of the main concepts in COP are behavioral variations that express partial definitions of the basic behavior of systems. In other words, they can be interpreted as portions of code that will be activated at runtime to reflect changes in context.

Behavioral variations are often cross-cutting concerns that are scattered throughout the code base. COP addresses this by providing a layer for grouping related behavioral variations so that they can be modularized. This eases the development and maintenance phases of applications. However, these behaviors must be specified at design time (Keays & Rakotonirainy, 2003). Behavior depends on users' needs and preferences, as well as changes over time. For this reason, it can be difficult and inefficient, to try to anticipate all possible behaviors at design time. Given this, the possibilities of adaptation of the software would be limited to a certain number of predefined contexts in the source code.

There are different techniques proposed for the activation of variations, which also deal with behavioral consistency since the activation of different variations can lead to conflicts in system behaviors. Currently, COP consists of a family of languages developed specifically to support context-aware adaptation, with some widely adopted design solutions and many different variants (Salvaneschi, Ghezzi, & Pradella, 2012). Some research in this field is devoted to realizing the COP paradigm by integrating COP constructs into an existing language, e.g. ContextJ (COP + java), ContextL (COP + Lisp) , ContextR (Ruby), ContextS, ContextLua, PyContext, etc (Appeltauer, Hirschfeld, Haupt, Lincke, & Perscheid, 2009). The main characteristics of COP are:

- COP is independent of how source code is organized into textual modules.

- Layers as named first-class entities that can be referred to explicitly at runtime, and whose composition can be dynamically controlled on-demand.

- It can be beneficial to activate/deactivate layers from anywhere in the code.

Thanks to COP, the foundations were laid for a new paradigm that would allow us to adapt the software to the context, and today, the smart devices to the characteristics of the context. COP facilitates the development of the Situational-Context due to it offers a vision of how the context-dependent software should be implemented and, therefore, this same methodology is applied to define the behavior of the devices in IoT environments.

2.3.2 Semantic Web

Describing information about people and devices in a common way facilitates their interaction. One of the most relevant technologies to favor the communication heterogeneous in IoT environments is the Semantic Web. This concept was coined by Tim Berners-Lee (Berners-Lee, Hendler, & Lassila, 2001). The Semantic Web is a mesh of data that is associated in such a way that it can be easily processed by machines rather than by human operators. It can be conceived as an extended version of the current World Wide Web and represents an effective means of representing data in the form of a globally linked database (Techopedia, 2017). Figure 2.3 represents the Semantic Web stack, where each layer uses the capabilities of the lower layers. It also shows how different technologies are organized to enable the Semantic Web. Similarly, technologies from the bottom of the stack to OWL (Figure 2.3) are currently standardized and accepted for building Semantic Web applications. To achieve a complete view of the Semantic Web in a system, all layers of the stack must be implemented.

The Semantic Web is powered by the World Wide Web Consortium (W3C). It is based on the Resource Description Framework (RDF) of the W3C and is usually designed with syntax using Uniform Resource Identifiers (URI) to represent data. These syntaxes are known as RDF syntax. The inclusion of data in RDF files allows software or web spiders to search, discover, collect, evaluate, and process data on the web. The key objective of the Semantic Web is to trigger the evolution of the existing Web to allow users to search, discover, share, and merge information with less effort. Humans can use the Web to perform multiple tasks, such as booking tickets online, searching for different information, using online dictionaries, and so on. Even so, machines cannot perform any of these tasks without

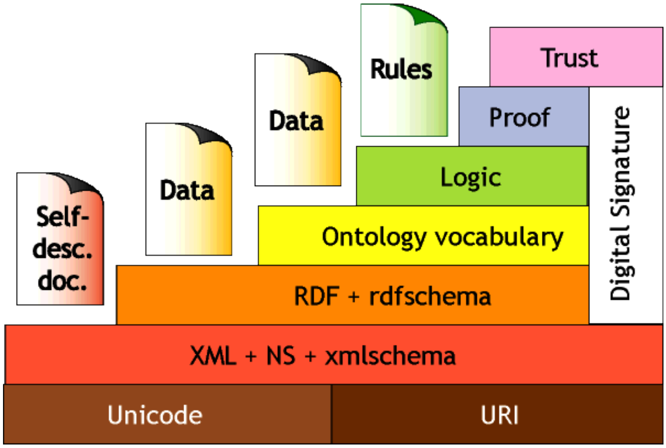


Figure 2.3: Semantic Web stack (Berners-Lee, 2000)

human intervention because web pages are made to be read by humans, not by machines. The Semantic Web can be considered a vision of the future in which data can be quickly interpreted by machines, allowing them to perform numerous and tedious tasks related to the discovery, mixing, and action of the information available on the Web. The Semantic Web research community has developed different languages for knowledge representation and reasoners for inferring its data. Some of the most popular languages are RDF/RDF-Schema (RDF-S), DARPA Agent Mark-up Language (DAML)/DAML-Service (DAML-S), or Web Ontology Languages (OWL)/OWL-Schema (OWL-S). Also, some of the most used semantic reasoners are: DLP, FaCT, RACER, Pellet, HermiT, or JENA. Each reasoner has its own characteristics, such as the inference algorithm involved, the supporting logic, the degree of completeness of the reasoning, or the implementation language used. Semantic Web languages can also be characterized by their expressive power, their ability to represent semantics, the constructs for knowledge representation, the underlying logic, and so on (Mishra & Kumar, 2011).

Therefore, the Semantic Web is a process that allows machines to understand and react quickly to complicated human demands based on their meaning. This understanding requires appropriate information sources to be semantically structured, which is a difficult task (Techopedia, 2017).

Semantic Web techniques have a great utility to relate devices that, having common characteristics, are located in different sources of data. Recent research is making a great effort to improve the relationship and communication between intelligent devices of the IoT. Klush et al. (Klusch, Kapahnke, Schulte, Lecue, & Bernstein, 2016) reviewed the state of the art where the importance of the Semantic Web in terms of search and device discovery is revealed, as well as the relationship between them. In (Gyrard, Datta, Bonnet, & Boudaoud, 2015b), some of the most important challenges of IoT, in semantic terms, are addressed: integrating semantic technologies, providing device interoperability, interpreting data, and facilitating the development of IoT applications. Besides, (Ruta, Scioscia, Loseto, & Di Sciascio, 2017) is inspired by social interactions to achieve greater self-configuration and self-orchestration for an intelligent building. The Semantic Web allows devices and subsystems to be able to determine a situation or discover other devices for data exchange. Semantic reasoners and algorithms are also vitally important within the Semantic Web. Scioscia et al. (Scioscia et al., 2014) developed a mobile inference engine, Mini-ME, which aims to discover resources and services in mobile and ubiquitous contexts. Also of special attention is the work done by Luiz H. Nunes et al. which proposes an algorithm for the search and discovery of heterogeneous resources in large-scale environments for reuse by other applications (Nunes, Estrella, Perera, Reiff-Marganiec, & Delbem, 2018).

As we can see, the Semantic Web takes on special relevance within the IoT and its use is widespread in this field. For this thesis, from the stack of layers shown above the ones we are most interested in to cope with data representation and data processing are ontologies and semantic reasoners. These elements allow for defining and storing information, and inferring knowledge from it, respectively, and contribute to the development of the Situational-Context. These two elements are introduced in the following sections.

2.3.2.1 Ontologies

The elements that define an ontology, such as its specification or stored data, are crucial for the understanding of the information by machine language. Ontologies are represented as metadata schemas that, through a previously defined vocabulary, can process and reason about the information to be used later by machine language. Ontologies are useful in a multitude of application domains and help people and machines to communicate concisely and provide semantic content to the information exchanged to facilitate its understanding. This is why the success and evolution of the Semantic Web depend to a large extent on the good usability and reusability applied to each domain of action (Maedche & Staab, 2001).

In terms of specification, ontologies can be formalized in different languages. The most popular language adopted by the community is the Web Ontology Language (OWL) standard of the W3C (World Wide Web Consortium). To use such languages, data needs to be labeled concerning an ontology specification to be shared between different parts of a system (machine language and people) in a way that preserves meaning and allows efficient queries.

Technically, an ontology consists of a set of concepts or classes that represent a domain of action. In addition, a set of defined relationships can be established over these concepts to extend knowledge and perform queries. These relations can be taxonomic and non-taxonomic (Maedche & Staab, 2001). On the one hand, taxonomic relations establish a hierarchical order between concepts or classes, which is semantically defined through the inclusion in the set of individuals of the class itself. On the other hand, non-taxonomic relations do not serve to order concepts hierarchically. Different tools are classified based on their features, editing, annotation, or querying, such as Protégé, Neon, WebOnto, Ontolingua, etc (Khamparia & Pandey, 2017).

Ontologies can be used in different application domains such as smart homes, transportation, energy efficiency, healthcare, or any other. One of the most important classifications is the one made by Uschold and Gruninger in (Uschold & Gruninger, 1996), where the following categories are established:

- **Communication between people and organizations**: They specify properties where different people must have a shared understanding of the system and its goals. Thanks to ontologies, a normative model of the system can be defined, which allows specifying semantics for the system and an extensible model that can be refined later. This classification also states that ontologies can be used to create a network of relationships, keep track of what is linked, and explore and navigate this network of data. Also, one of the most important functions of ontology in communication is that it provides unambiguous definitions of terms used in a software system.

- **Interoperability between systems**: This classification is of special interest since one of the objectives addressed in this thesis is the problem of interoperability between systems. One of the objectives of ontologies is to favor the creation of easily integrated environments for different software tools and people. IT-dependent environments must use integrated business models that encompass domain-specific activities, resources, products, and services. These integrated business models are useful as a common repository that can be accessed by different types of systems. In addition to systems and repositories, several distinctions can be made. First, the nature of the relationships between users sharing tools and data must be considered. It is vital that ontologies and tools used by different agents or organizations within the same enterprise be shareable and reusable among these multiple organizations.

- **Systems engineering to specify data, reuse components and improve reliability**: The role played by ontologies varies according to the degree of formality and automation within the methodology used for system design. In this regard, we differentiate between informal and formal approaches. In an informal approach, ontologies facilitate the process of identifying system requirements and understanding the relationships between its components. This is essential from the point of view of systems involving teams of people from different domains. In a formal approach, ontology provides a declarative specification of a software system. This allows reasoning about what the system is designed to do, rather than how the system supports this functionality. One of the most important aspects of ontologies is reusability. Ontologies should also enable reuse so that developers or users can easily import and export modules between different systems. However, when software tools are applied to new domains, it is common that they do not work as expected, since they are based on systems for which they were originally designed, but not on the new ones. To achieve a high degree of reusability, domain classes and tasks within these domains are characterized. In this way, ontologies provide a framework for determining which aspects of an ontology are reusable between different performance domains.

Therefore, the use of ontologies to favor the integration of systems and people data has many advantages, thanks to the vocabulary by which they are formed and the relationships that allow efficient searches. Moreover, they provide an easily interpretable way to represent and share knowledge using a common vocabulary, through a knowledge exchange format, and allow knowledge reuse. This is why ontologies are a very interesting option to consider for integrating different systems with people, and they consist of covering the semantic and syntactic gaps between different data sources to favor communication.

The use of ontologies is an aspect that will be present throughout the development conducted during this thesis. Thanks to them it is possible to define the information of people and devices. This definition also helps us to store the information in order to be able to infer knowledge when identifying situations in the Situational-Context.

2.3.2.2 Semantic Reasoners

Typically, systems developed for the Semantic Web require some kind of reasoning capability. It was discussed that for knowledge representation, ontologies and their correctness play an important role. To ensure the quality of the information described from ontologies, it is necessary to address the potential problems of information inconsistency, and uncertainty of data from real-world information sources. Because of this, results cannot be correctly represented and interpreted, leading information systems to incorrect semantic understanding and representation. Semantic reasoners, in this case ontology-oriented, reduce data redundancy and information error in the knowledge base and also detect possible content conflicts. Moreover, they can infer knowledge based on the information described by the ontology, thus enriching the represented information (Khamparia & Pandey, 2017).

Thus, a semantic reasoner is a tool that infers logical consequences from facts, assertions, and axioms that provides support for automated reasoning (Khamparia & Pandey, 2017). There are currently numerous semantic reasoners such as Pellet, FaCT, HermiT, CEL, Cerebra, or JENA. These reasoners are distinguished based on their features and construction methods. Usually, when selecting a reasoner, the methodology, robustness, completeness, support of inferences, type of reasoning obtained, and implementation performed are studied (Khamparia & Pandey, 2017).

The use of semantic reasoners in the Situational-Context allows us to infer the knowledge necessary to identify situations in smart environments. This knowledge is based on the information provided by people and devices, the processing of which is crucial to identify what situation is occurring at a given time. In addition, reasoners help in predicting possible future situations in order to facilitate the adaptation of devices more quickly.

2.3.3 People as a Service

The above paradigms have helped us understand that smart environments must be people-centric and devices must meet their preferences. In this sense, we find the People as a Service (PeaaS) paradigm (Guillen et al., 2013), which aims to take advantage of the potential of mobile devices to store users' sociological profiles and to offer them as services to other devices. PeaaS is characterized by maintaining the integrity of user data captured from the environment through their mobile devices and using it to adapt to the environment. Specifically, PeaaS is a mobile-centric model to detect and infer the context of smartphone owners and generate their sociological profiles. These profiles collect valuable information to identify moods, trends, or health habits, creating digital projections of their owners, and are stored and shared from the owners' mobile devices. In addition, profile owners can decide with whom and for what purpose to share their sociological profiles, ensuring that privacy is fully in their hands. Many applications with human data typically require much more complicated calculations to determine specific actions, while applications based on the PeaaS architecture typically rely on a smaller set of data and associated calculations. Therefore, PeaaS is a sufficiently relevant solution to establish a design pattern to facilitate its implementation.

Therefore, PeaaS is based on four fundamental pillars to secure user data and to be able to offer it as a service:

- **Mobile devices as interfaces for people**: these devices connect users with the outside world, being able to be used to express emotions, thoughts, or concerns.
- **Sociological virtual profiles**: people's behavior in certain situations can be detected and analyzed. This information is stored in the virtual profile of the device owner, together with contextual information captured by device sensors and interactions with other people and devices.
- **Sociological profiles as a service**: once a person's sociological profile is generated, it can be offered as a service. According to the PeaaS philosophy, sociological profiles should be offered as a service to those devices and people who wish to access this information, always bearing in mind that it is the owner of the profile who decides to whom to give access and to what data.
- **User privacy**: it is guaranteed that a person's sociological profile is always kept on his or her device. All-access to a profile is visible and controllable by the owners of the device, allowing its monitoring and controlling who accesses the information, when, and where.

The authors designed a reference architecture to ensure compliance with the above 4 pillars (Figure 2.4). This architecture is based on people's mobile devices, which contain the necessary components to comply with the PeaaS model. Although initially proposed as an architecture on the application layer, the approach could be exploited by a development closer to the operating system or hardware, where the information exchange is performed through the capabilities of the mobile device.

The first pillar (*mobile devices as interfaces for people*) is met through the development of an inference engine that relates people as interfaces to their mobile devices, to facilitate interaction with external entities. For these interactions the model evaluates whether the owner of the device meets a set of criteria, such as gender, age, or interests, to determine

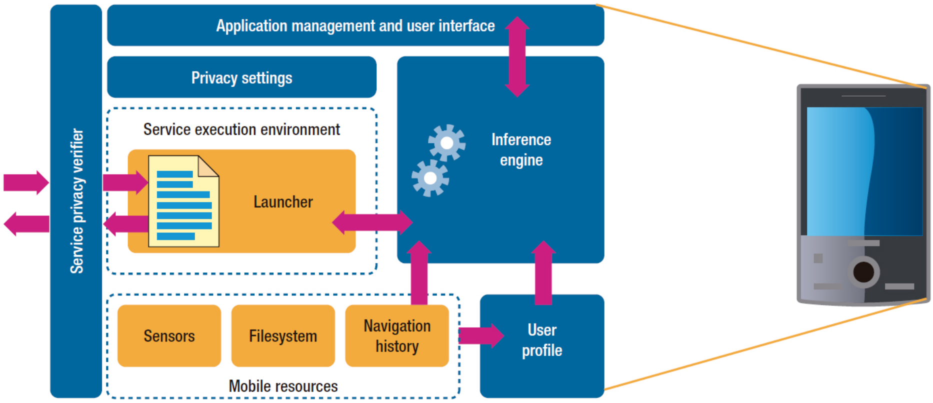


Figure 2.4: PeaaS reference architecture (Guillen et al., 2013)

whether incoming interactions are eligible or not. This ensures active and interactive communication with the device owner.

For the second pillar (*sociological virtual profiles*), users have a sociological profile where all their information is stored as a virtual identity. This profile should monitor the capabilities of the device to which it is associated and the interactions of its owner, store all information securely and locally, and allow it to be used to generate and add new information.

Next, the third pillar (*sociological profiles as a service*) allows services to be deployed from the owner's own mobile device and allows external entities to consume these services. These services are born from the sociological profile of their owners and aim to encourage interactions between entities in the environment. To do this, the authors propose that the services be provided using standard communication protocols such as REST or SOAP.

Finally, for the fourth pillar (*user privacy*) a component is proposed that contains the privacy and security policies of the device and, therefore, of its owner. This addresses what services can be deployed, under what conditions, or who can consume them. Therefore, it is the profile owners who store their own information and decide with whom to share it through different types of services.

PeaaS paradigm is being widely used by the scientific community, either to develop architectures, IoT applications or to emphasize the importance of treating people as service providers (Miranda et al., 2015; Berrocal, Garcia-Alonso, Murillo, & Canal, 2017; Pérez-Vereda & Canal, 2017; Laso, Berrocal, García-Alonso, Canal, & Manuel Murillo, 2021).

The management proposed by PeaaS to offer people's information as a service is an interesting aspect in the development of Situational-Context. Situational-Context considers that both people and smart devices can offer services. Therefore, the methodology proposed by PeaaS around its fundamental pillars helps us to define how the services offered by people should be defined and what information is necessary to make this possible.

2.3.4 Ambient Intelligence

The analysis of the environment has also been the subject of study so that devices can adapt to people's needs. AmI paradigm seeks to make everyday environments responsive and adaptive to people. In this paradigm, the word *ambient* refers to the integration of technology in a non-intrusive way into everyday devices and environments, while the word *intelligence* indicates that digital environments present specific characteristics for social interaction. That is, environments can recognize people, identify their preferences, and adapt to them while in the environment. AmI environments are sensitive to the needs of their users. This helps anticipate people's preferences and behaviors. In addition to being aware of this, these environments can interact with people in a friendly way and even recognize and respond to emotions or certain stimuli.

In terms of device distribution, an AmI environment is based on miniaturized, low-cost hardware, which provides complex networks of heterogeneous information appliances or smart artifacts. These, either individually or collaboratively, enable people's daily tasks to be simplified or automated. Thus, energy consumption can be reduced by controlling lights and blinds automatically, a higher level of security can be achieved by having devices that send alerts in emergencies, or simply improving the quality of life and the level of comfort with devices created especially for this purpose.

There are many settings in which AmI can greatly impact our lives. Sample areas of AmI application were extracted from (Cook, Augusto, & Jakkula, 2009).

- **Smarthome:** Several artifacts and items in a house can be enriched with sensors to gather information about their use and in some cases even to act independently without human intervention. Some examples of such devices are appliances (e.g., cooker and fridge), household items (e.g., taps, bed, and sofa), and temperature handling devices (e.g., air conditioning and radiators).

- **Health monitoring and assistance:** With the maturing of supporting technologies, at-home automated assistance can allow people with mental and physical challenges to lead independent lives in their own homes and reduce the physical and emotional toll that is taken on caregivers. Some of these technologies focus on assurance or making sure our friends and loved ones are safe and healthy at home. AmI techniques can be used to provide reminders of normal tasks or the sequence of steps that comprise these tasks

- **Hospitals:** Applications of AmI in hospitals can vary from enhancing safety for patients and professionals to following the evolution of patients after surgical intervention. Many of the AmI technologies found in smart homes can be adapted for use in specific rooms or areas of a hospital. E.g, Patients are entertained and helped by AmI during their examination sessions or services by monitoring patients' health and progress through analysis of activities in their rooms.

- **Transportation:** Transport means are also valuable settings for AmI technologies. Train stations, buses, and cars can be equipped with technology that can provide fundamental knowledge about how the system is performing at each moment. Public transport can benefit from AmI technology including GPS-based spatial location, vehicle identification, and image processing to make transport more fluent and hence more efficient and safe.

- **Education:** Education-related institutions can use technology to track students' progression on their tasks and the frequency of their attendance at key events. E.g., human-computer interfaces through devices such as an interactive whiteboard that stores content in a database, or a smart classroom where the experience is enhanced by video and microphones that recognize a set of gestures, motions, and speech that can be used to retrieve information or focus attention on appropriate displays and material.

- **Workplaces:** The design of intelligent workspaces, conference rooms, and kiosks that use a variety of mechanisms such as gaze-aware interfaces and multi-modal sketching that the full meaning of a discussion between co-workers through enhancing the performance of the employees at work.

As we can see there are multiple areas of application where AmI has great interest. The goal of AmI is not only to provide such active and intelligent technologies but to weave them seamlessly into the fabric of everyday lives and settings and to tailor them to each individual's specific needs. This aspect is closely related to one of the objectives of this thesis, which aims to achieve the resolution of people's needs through smart devices. However, AmI needs prior knowledge of user preferences to establish when a device should act. This is a complex problem when the needs of several people have to be analyzed, especially when their presence was not originally foreseen in the system.

AmI is a paradigm that has been with us for many years and is still being studied today to develop new smart ecosystems and also serves as an inspiration for the development of others, such as the Internet of Things. That is why in this thesis, the AmI paradigm was one of the first to be explored to have the fundamentals of how smart environments work, what characteristics they usually have, and how smart devices can satisfy people's needs, which suppose a great contribution to developing the Situational-Context.

The analysis of these areas related to the pillars of the Situational-Context and identified the contributions they can make to its development, we consider it necessary to review the most relevant work in these areas. This review will be based on identifying what works exist in the literature on the integration of devices, identification of situations, and adaptation of IoT environments to the context. This is done in the following section by developing a systematic literature review.

2.4 Systematic Literature Review

As we have seen, there are numerous areas related to the development of Situational-Context. In the previous section, these areas were introduced to provide the necessary background on how they relate to the objectives of this thesis. Once the necessary knowledge is available, the next step is to search for the current work in the literature on these areas. For this purpose, a Systematic Literature Review (SLR) was performed.

A SLR is a type of scientific research whose objective is to objectively and systematically integrate the results of empirical studies on a given research problem (Kitchenham & Charters, 2007). There are several methodologies to follow to conduct a SLR. For this thesis, the methodology defined by Kitchenham et al. in (Kitchenham & Charters, 2007) was selected, where a series of steps must be conducted to achieve the proposed objective, from the initial identification of related works to the exhaustive review of the most relevant and related to the analyzed area. Each of these steps will be described in the following sections.

2.4.1 Objectives

We start from the premise that a smart environment is made up of people and devices of all kinds, and that through collaboration between them, proactive behaviors are produced that allow adapting the environment to the needs of people.

Thus, the objective of this SLR is to identify the works that, first, address the IoT as a smart ecosystem, second, favor communication between devices with different characteristics, third, refer to context-aware systems, fourth, propose solutions based on the development of architectures, platforms, or specific systems, and, fifth, that these developments favor adaptation to the needs of people.

2.4.2 Research Method

According to Kitchenham et al. (Kitchenham & Charters, 2007), the research method to conduct an SLR consists of three main activities: protocol planning, SLR execution, and results reporting. Protocol planning consists of drawing up a work plan that allows each of the necessary steps to be carried out in an orderly manner. The execution of the SLR is the main activity that consists of searching, analyzing the work, and extracting the necessary and relevant data for the study. The results report provides the results of the study in the form of tables, graphs, or any other medium that allows the results obtained throughout the study to be appreciated simply and clearly.

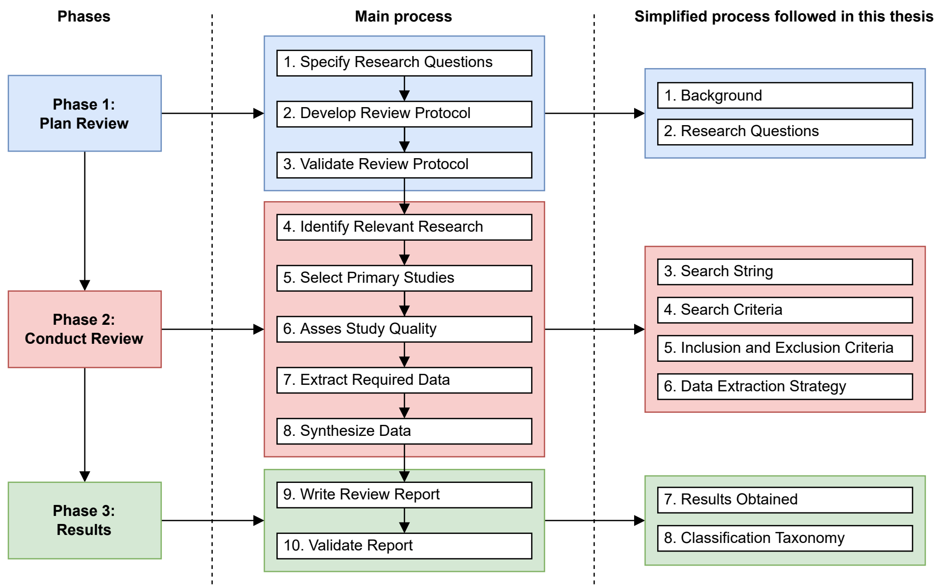


Figure 2.5: SLR process followed

Figure 2.5 shows each of these phases. For the development of this methodology, some of the sub-activities were unified in this thesis to simplify the process. The most relevant phases for SLR are shown below, and the whole process is shown in Appendix I.

2.4.3 Research Questions

The research questions are one of the most crucial parts of SLR development. They must be specified correctly since they condition the rest of the work. Therefore, it is necessary to establish the right questions to solve the detected needs. Table 2.1 specifies the research questions posed for this SLR.

Table 2.1: SLR Research Questions

#	Research Question	Motivation	Related to Need
RQ1	What technologies exist for managing IoT/WoT smart environments?	Know the existing applications or frameworks that allow connecting IoT devices to interact with the environment. There are platforms that allow installing devices that, through a previous configuration, can automate routines or make changes in the environment. For example, using a light bulb with Alexa.	N1
RQ2	What standards exist to provide interoperability between IoT devices?	Know the most popular protocols for exchanging information between devices. There are many communication and information exchange protocols on the market. This heterogeneity allows having a wide range of devices, but on the other hand it makes compatibility between devices difficult. For example, using the Semantic Web.	N2
RQ3	What architectures/middlewares exist to provide interoperability between IoT devices?	Know the techniques used in existing works that favor interoperability. The problem of heterogeneity in IoT has been addressed in many existing works. We aim to analyze these works to learn about the techniques employed and how they can be applied in the context of this thesis. For example, using Home Assistant or Open Hub.	N2
RQ4	What kind of interoperability do current solutions provide?	Know the types of interoperability addressed in each paper. There are many levels of interoperability and the current works tend to focus on providing solution to them individually. The aim is to analyze the works that address these levels individually or jointly and how they are treated in IoT environments. For example, interoperability at the level of connectivity and information exchange, or at the semantic level.	N2/N3
RQ5	What techniques are used to adapt devices to people?	Know what techniques are used for devices to adapt their behavior. There are many elements that affect the adaptation of an IoT environment, such as the context, nearby devices, the people involved, etc. The aim is to analyze how these elements are treated to decide the behavior of a device to a given event. For example, setting the brightness of a light bulb.	N3

2.4.4 Search Criteria

Once the research questions were established, the next step is to establish the search criteria. To do this, the sources of scientific resources to be searched must be selected. Among the most famous sources are Web of Science (WOS) [1], Google Scholar [2], or Scopus [3]. For this SLR we have decided to use **Scopus** because it is one of the most complete and relevant sources for locating papers related to Computer Science.

2.4.5 Search String

The search string will depend on the terms we want to locate in the papers. In addition, alternative terms can be entered for each term to cover a wider range of possibilities. The terms used for this SLR are listed in Table 2.2.

Table 2.2: SLR Search string composition

Main Term	**Alternative Terms**	**Related to Question**
"Internet of Things"	IoT OR "Web of Things" OR WoT OR "Smart Environment"	RQ1
Interoperability	Interconnection OR Connectivity OR Communication OR Protocols	RQ2/RQ4
Context	Situation OR "Situation Aware" OR "Situation-Aware" OR "Context Aware" OR "Context-Aware"	RQ4
Architecture	Platform OR Middleware OR Framework OR Standard	RQ2/RQ3
Adaptation	"Self-adaptation" OR Behaviour OR Needs OR Preferences OR People	RQ5

As can be seen, papers are desired that discuss the Internet of Things, interoperability, context, architectures, and adaptation, each of these terms with their respective alternative terms. From these terms, we can establish the search string that will allow us to locate the jobs to be analyzed. The search string goes through several modifications until the final one is decided. In addition, a filter was used to limit the search area to computer science (COMP).

[1] https://www.webofscience.com/wos/woscc/basic-search
[2] https://scholar.google.com
[3] https://www.scopus.com

2.4.6 Results Reporting

2.4.6.1 Classification Taxonomy

The analyzed papers provide a broad understanding of IoT interoperability. Each of them has its focus on a specific topic. Whether it is improving interoperability at the device level, providing specific communication protocols, at the semantic level, reusing ontologies, or developing a framework or architecture that encompasses several of these aspects. For this reason, the most relevant information was extracted from each of them to classify them into different areas. With the identified areas, and with the results obtained from the questions, the answers, and the extraction form, a taxonomy was developed to identify the main interests of the research community and where the problem of interoperability in context-oriented IoT is heading. Figure 2.6 shows the taxonomy based on the analyzed papers.

The main concerns are related to the technologies used to connect devices, the application domains where IoT is integrated, how the semantic web is used to improve the relationships between devices, how different situations can be identified in the environments, how context awareness affects the creation of IoT applications, how the syntax of information is defined to be treated in a common way, and some other issues such as best practices or compilations of relevant work. Within these concerns, a subdivision has also been made to classify a finer grain in the analyzed works. The following section shows the results obtained according to this classification taxonomy.

2.4.6.2 Results Obtained

This section presents the results obtained from the SLR conducted on the **52** papers finally included in the analysis (Phase 4). These results are obtained by filtering through the different phases of the process until the final phase is reached. Figure 2.7 shows this process and Appendix I details each of these phases.

Below the results are analyzed. Firstly, the general indicators that deal with the origin of the papers are detailed, and, secondly, the specific indicators that refer to the classification taxonomy used are shown.

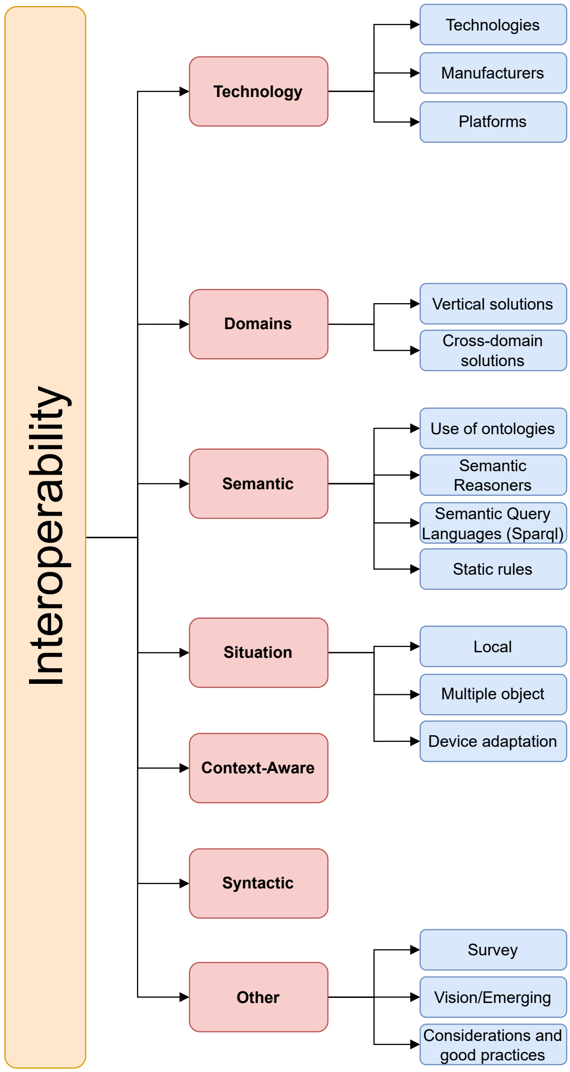


Figure 2.6: Classification taxonomy

General indicators The first indicator to be analyzed refers to the origin of the papers (Figure 2.8). It can be seen that 39 papers came directly from the Scopus search (76.47%) and another 12 from other sources (23.52%),

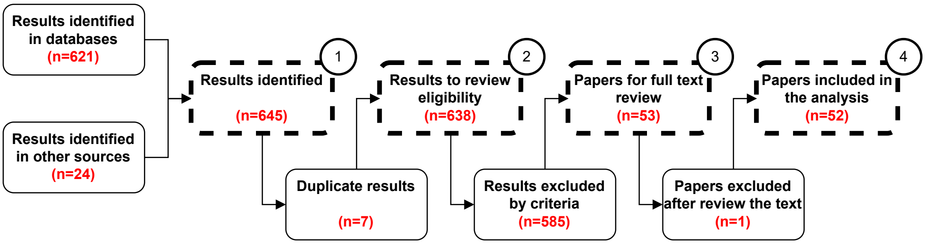


Figure 2.7: SLR process summary

which correspond to papers already identified as relevant before starting the SLR. This indicates that most of the papers identified were included in the search chain, but we still analyzed papers that were previously selected.

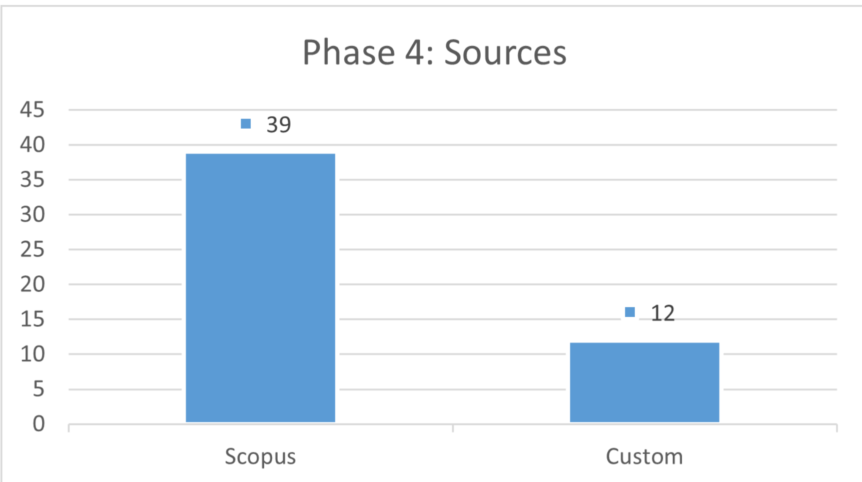


Figure 2.8: Main sources

The second indicator refers to the type of paper (Figure 2.9). The types to be considered were conferences, journals, books, or books of proceedings. It can be seen that 21 papers come from conferences (40.38%) (mostly indexed), 27 from impact journals (51.92%), 2 from books (3.84%), and 2 from books of proceedings (3.84%). This indicates that most of the papers come from conferences and impact journals, which would allow us to satisfy the research questions with a fairly accurate criterion.

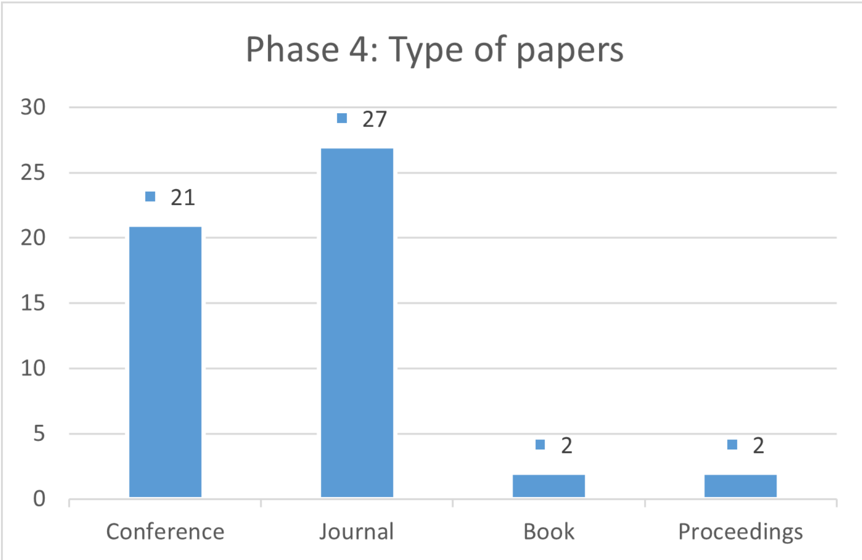


Figure 2.9: Type of papers

The third indicator refers to the year of publication of the papers analyzed (Figure 2.10). It can be seen how most of the papers are concentrated in 2015 from where 9 (17.31%), 2016 from where 7 (13.46%), 2017 from where another 9 (17.31%), and 2019 from where 8 (15.38%) originate. This indicates that most of the developments on IoT interoperability are coming from the last few years.

The general indicators show that most of the articles analyzed come from conferences and journals. In addition, it also shows that interest in addressing the challenge of interoperability has been growing over the years and is increasingly being addressed by more and more researchers.

Specific indicators Each of the papers reviewed has been included in the classification taxonomy detailed above. In addition, a paper can be included in several categories, so it does not have to belong exclusively to one. Figure 2.11 shows the distribution of the papers. It can be seen how 34 of the papers (64.15%) deal with improving interoperability at the tech-

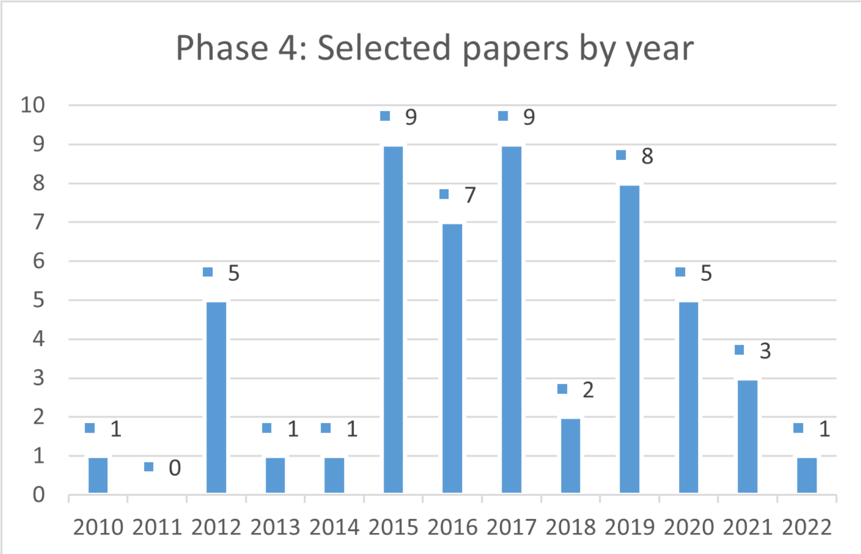


Figure 2.10: Papers by year

nological level, 20 papers (37.74%) address interoperability issues from the context-aware perspective, 18 papers (33.96%) focus on semantic interoperability, and 21 papers (39.62%) address other types of interoperability or are compilations or surveys of existing papers.

In addition, the analyzed papers have also been included in the subdivisions made for each level of interoperability to know more precisely the degree of affinity with the level where it is included. This makes it possible to obtain more specific indicators. These indicators will be detailed below for each of the levels of interoperability analyzed.

Figure 2.12 shows the classification of papers dealing with technological interoperability. It can be seen how 21 of these papers (45.65%) focus on technologies in general, 4 papers (8.70%) deal with actions taken on manufacturers, and the remaining 21 papers (45.65%) deal with the use of different platforms.

Figure 2.13 shows the results obtained for papers addressing IoT domain interoperability. Of the two subdivisions identified, 5 papers (50.00%) address vertical solutions that attempt to cover many different domains,

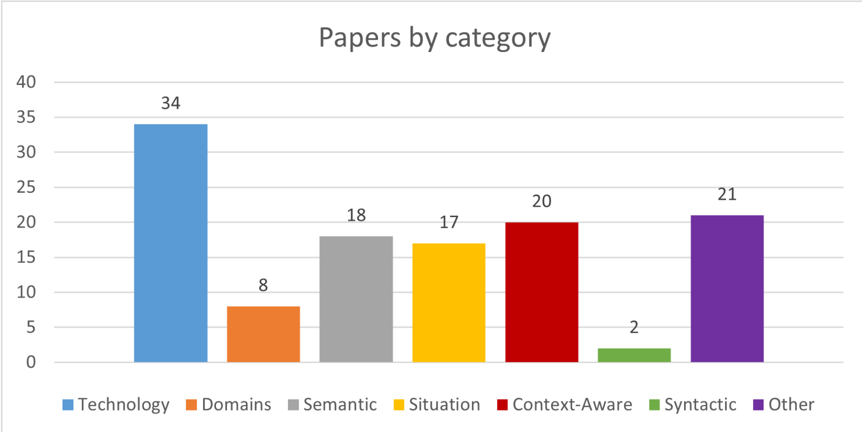


Figure 2.11: Papers by category

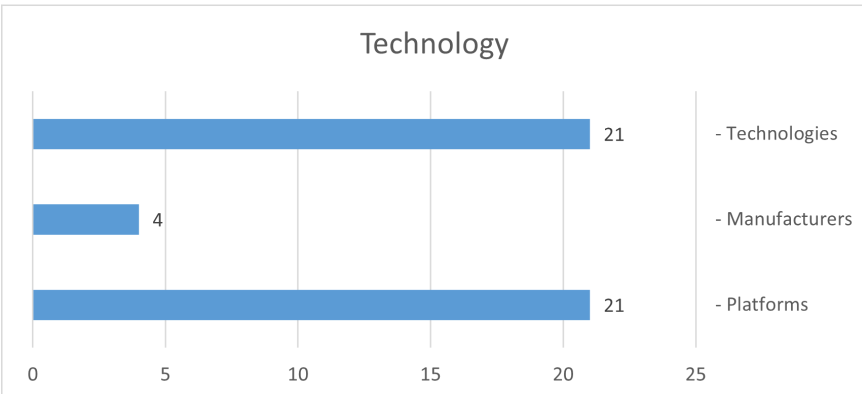


Figure 2.12: Technological category

and 5 papers (50.00%) address horizontal solutions that focus on covering all possible aspects of a single application domain.

Figure 2.14 shows the results obtained for the semantic domain. The most important division is the use of ontologies, where 16 of the analyzed papers (53.133%) addressed this topic. The use of semantic reasoners was addressed in 6 of the papers (20.00%). The remaining divisions compose 4 of the papers (13.33%).

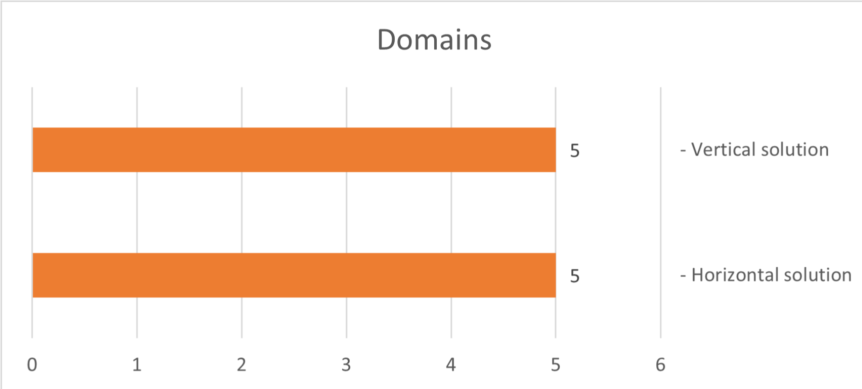


Figure 2.13: Domains category

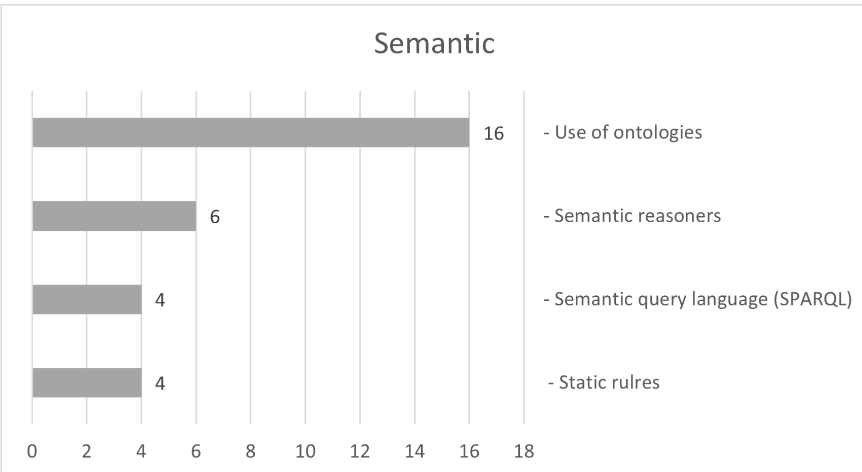


Figure 2.14: Semantic category

Figure 2.15 details the results obtained within the classification of the level related to situation identification. Of these papers, 12 papers (38.71%) focus on situations where multiple IoT objects are encountered, 12 papers (38.71%) address the adaptation of devices to the situation, and another 7 papers (22.58%) perform processing to identify local situations.

Figure 2.16 shows the results obtained from the articles that consider

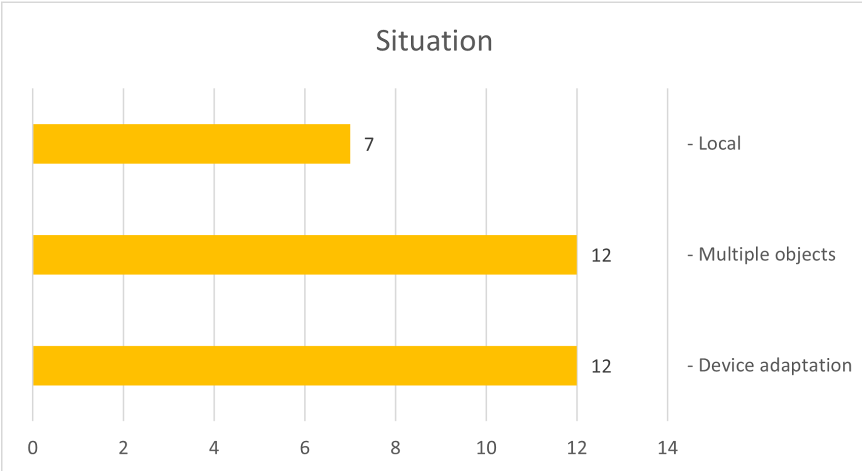


Figure 2.15: Situational category

context awareness to improve IoT interoperability. The 20 articles identified (100.00%) are of a general nature within the context treatment for IoT environments.

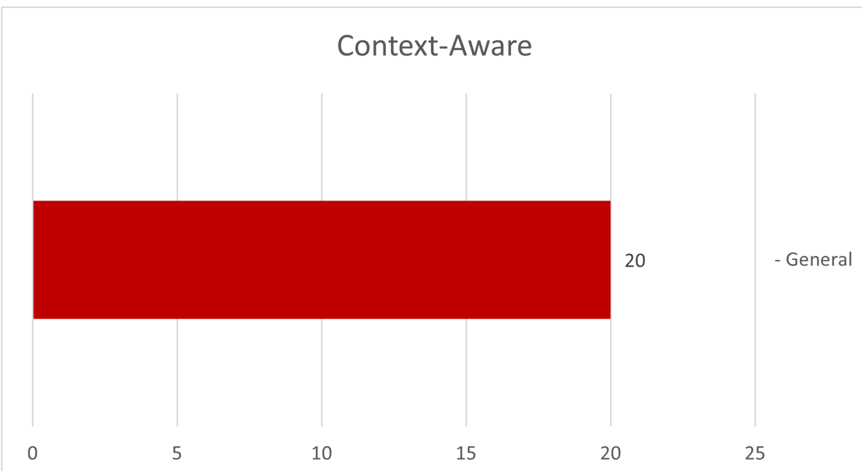


Figure 2.16: Context-Aware category

Figure 2.17 shows the results obtained at the syntactic level. As in the previous level, also insufficient details were found to perform several subdivisions. The 2 papers (100.00%) identified that address this domain deal mainly with the packaging and transmission of information within IoT environments.

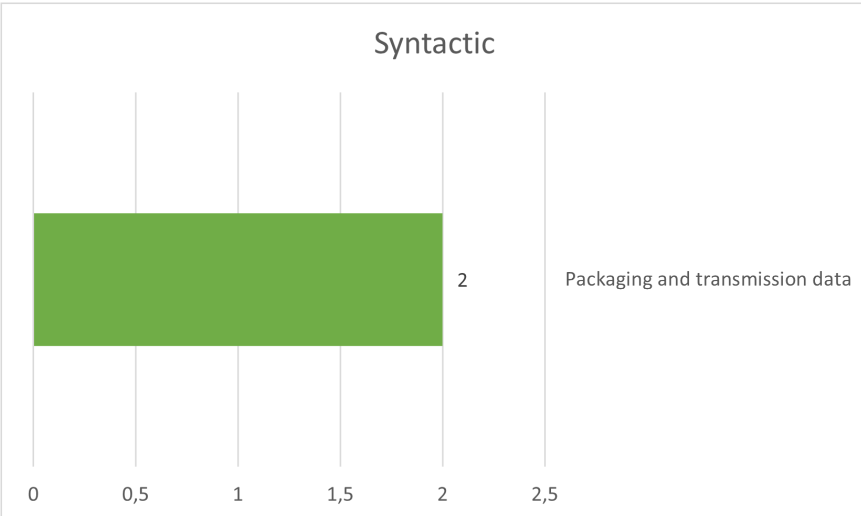


Figure 2.17: Syntactic category

Figure 2.18 shows the last classification, where other types of papers are grouped. At this level, several subdivisions have been made, where 14 papers (48.28%) correspond to surveys that compile or summarize other related papers, 9 papers (31.03%) refer to emerging solutions, and 6 papers (20.69%) deal with considerations to be taken into account and follow-up of good practices.

Because of the works analyzed, a gap is identified in the treatment of the context for IoT environments, more specifically in the identification and processing of the different situations that may occur. On their own, these works bring great benefits to IoT interoperability, either at a specific level of interoperability, using a particular standard, or focused on a specific application domain. However, it is the combination of some of the aspects addressed by these papers that motivates this doctoral thesis, and that is

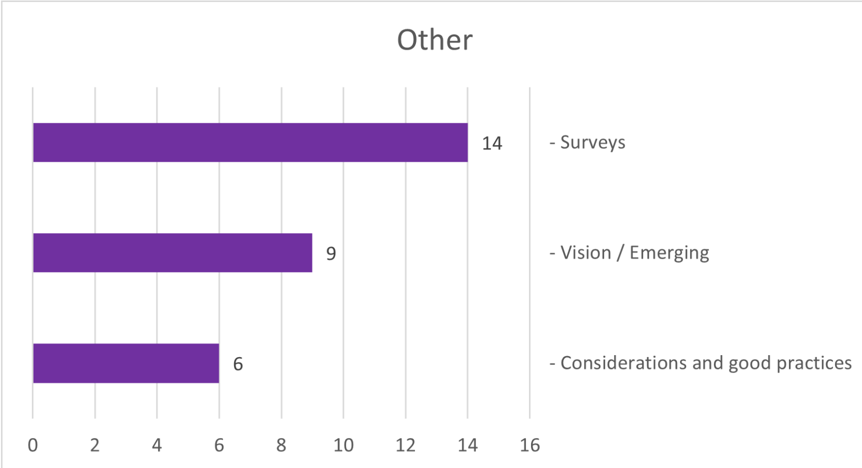


Figure 2.18: Others categories

where the idea of identifying situations in IoT environments to adapt the behavior of devices to the needs of people is born. The works analyzed and the levels addressed by each of them according to the previous taxonomy are detailed in Tables 2.3, 2.4 and 2.5.

Finally, in the following section, we present the conclusions drawn from the SLR conducted.

Table 2.3: SLR Analyzed papers (1/3)

Level	**Paper**																	
	(Rakib & Uddin, 2019)	**(Bonino et al., 2015)**	**(Pötter & Sztajnberg, 2016)**	**(van der Schaaf et al., 2020)**	**(Amarilli et al., 2017)**	**(De Matos et al., 2017)**	**(Alessi et al., 2016)**	**(Sasirekha & Swamynathan, 2016)**	**(Sezer et al., 2018)**	**(Temglit et al., 2016)**	**(Khan et al., 2012)**	**(Catania et al., 2019)**	**(Noura et al., 2018)**	**(Hussain et al., 2015)**	**(Merkle & Zander, 2016)**	**(Flores-Martin et al., 2019)**	**(Javed et al., 2017)**	**(Čolaković & Hadžialić, 2018)**
Technology																		
- Technologies			✓	✓			✓					✓		✓			✓	✓
- Manufacturers				✓														
- Platforms			✓	✓							✓	✓	✓	✓				
Domains																		
- Vertical solution		✓														✓		
- Horizontal solution		✓		✓														
Semantic																		
- Use of ontologies	✓		✓		✓			✓					✓		✓	✓		
- Semantic reasoners	✓							✓				✓	✓		✓			
- Semantic query languages															✓			
- Static rules						✓							✓					
Situation																		
- Local																		
- Multiple objects				✓	✓					✓	✓	✓				✓		
- Device adaptation				✓	✓	✓	✓						✓			✓		
Context-Aware																		
- General							✓					✓		✓		✓		
Syntactic																		
Packaging and transmission data																		
Other																		
- Surveys									✓									✓
- Vision / Emerging											✓							
- Considerations and good practices																	✓	

Table 2.4: SLR Analyzed papers (2/3)

	Paper																	
Level	**(Yaqoob et al., 2017)**	**(Gomez et al., 2019)**	**(Al-Fuqaha et al., 2015b)**	**(Noura et al., 2019b)**	**(Venceslau et al., 2019)**	**(Vinob chander, 2010)**	**(Asghar et al., 2015)**	**(Mongiello et al., 2016)**	**(Motta et al., 2017)**	**(Tayur & Suchithra, 2017)**	**(Barnaghi et al., 2012a)**	**(Giordano & Spezzano, 2014)**	**(Cheng et al., 2017)**	**(Cheng et al., 2016)**	**(Pantsar-Syväniemi et al., 2012)**	**(Busold et al., 2015)**	**(Arcaini et al., 2020)**	**(Davoudpour et al., 2015)**
Technology																		
- Technologies		✓	✓	✓		✓		✓	✓	✓					✓			✓
- Manufacturers																		
- Platforms						✓		✓				✓	✓	✓		✓	✓	✓
Domains																		
- Vertical solution		✓				✓												
- Horizontal solution			✓															
Semantic																		
- Use of ontologies			✓	✓	✓					✓	✓				✓			✓
- Semantic reasoners											✓							
- Semantic query languages				✓											✓			
- Static rules															✓			
Situation																		
- Local													✓	✓			✓	✓
- Multiple objects													✓	✓	✓		✓	
- Device adaptation													✓	✓	✓			
Context-Aware																		
- General			✓			✓		✓		✓	✓	✓	✓	✓	✓	✓		✓
Syntactic																		
- Packaging and transmission data																		
Other																		
- Surveys		✓	✓	✓	✓				✓	✓	✓							✓
- Vision / Emerging	✓						✓											
- Considerations and good practices	✓						✓											

Table 2.5: SLR Analyzed papers (3/3)

Level	Paper																
	(Perwej et al., 2019)	(Thomas et al., 2015)	(Maheswaran & Misra, 2015)	(Maita-Tepán et al., 2019)	(Cubo et al., 2012)	(Said & Masud, 2013)	(N. Kim et al., 2015)	(Gusmeroli et al., 2010)	(Saputra et al., 2019)	(Ali et al., 2017)	(Ji et al., 2021)	(Adam et al., 2020)	(Yu et al., 2020)	(André et al., 2019)	(Ortiz et al., 2022)	(Abbasi et al., 2021)	(Pradeep et al., 2021)
Technology																	
- Technologies								✓			✓	✓	✓			✓	
- Manufacturers								✓			✓			✓			
- Platforms			✓		✓			✓	✓	✓				✓			✓
Domains																	
- Vertical solution								✓									
- Horizontal solution								✓								✓	
Semantic																	
- Use of ontologies										✓			✓				
- Semantic reasoners																	
- Semantic query languages										✓							
- Static rules																	
Situation																	
- Local										✓					✓		✓
- Multiple objects										✓					✓		
- Device adaptation										✓							✓
Context-Aware																	
- General								✓	✓	✓			✓				✓
Syntactic																	
- Packaging and transmission data			✓		✓												
Other																	
- Surveys	✓					✓		✓			✓						
- Vision / Emerging	✓	✓	✓			✓	✓	✓									
- Considerations and good practices	✓	✓				✓											

2.4.7 Conclusions

Conducting an SLR involves reviewing the state of the art in a given field to see what state it is in and where a contribution can be made. The SLR conducted in this doctoral thesis has made it possible to identify the areas where IoT is most focused in terms of interoperability and, in turn, those where efforts are still needed to make further progress.

The tables above detail the main concerns addressed by the current literature. The characteristics extracted from these papers have allowed us to develop a taxonomy so that we can have an overview of where each paper stands, what area it covers, and in which one's effort still needs to be expended. In view of the results obtained, we found that most of the work focuses on improving interoperability at the technological level. The domain and syntactic levels are the domains where less work has been located. The semantic and Context-Aware domains have also been widely addressed through the analyzed works. Finally, other types of work such as surveys or best practice guides have also focused on analyzing interoperability.

For this thesis, the most relevant level is the situational level. We find that existing works address this level of interoperability, mainly for the adaptation of devices in local environments. However, the analyzed works do not show concern for addressing this aspect in dynamic environments. This refers to those environments where dynamic collaboration occurs depending on the devices and people that are in those environments at any given time. Moreover, these dynamic environments have to deal with the information coming from these devices and people to be able to adapt conditions to people's preferences. Therefore, we think that the dynamicity of these environments is a potential gap to be analyzed. The analyzed works also provide us with different technologies that have been successfully used in related areas and that can be used to develop the Situational-Context, such as *MQTT* as a protocol message, *W3C Thing Description* to describe the device information, *Protege* to define ontologies, or *RESTful* to orchestrate services. With this thesis, we bring the knowledge acquired during the background analysis and propose a viable and functional solution to deal with this kind of dynamic environment to detect specific situations.

The following chapter details the Situational-Context paradigm and how it addresses this gap.

Chapter 3

Situational-Context: Introduction and Developed Architecture

"Creativity is intelligence having fun."
Albert Einstein

Contents

This section details the development conducted for the Situational-Context. First, the concept of Situational-Context is defined to provide a more detailed view of the paradigm; second, an architecture that provides support for the identification of situations through different components is designed; and, third, the implementation of this architecture is performed through a real case study.

3.1 Definition

The idea of Situational-Context was born back in 2016 as a vision article outlining the first ideas for identifying situations in IoT environments. From there, it began a line of research that developed into this thesis.

The Situational-Context is defined as the composition of the virtual profiles of all the entities involved in a situation. Each entity has a virtual profile which we call a description and which contains the following information:

- **Personal information**. It contains the raw information of the entity: name, associated device, preferences, activity history, relationships with other entities, etc. This information is dynamic and is updated as the entity interacts with other entities.

- The **goals** detail the state of the entity's desired environment. These goals can also be deduced from the basic information. For example, having the goal of setting a certain temperature to achieve the comfort level in a room.

- The **skills** that the entity has to make decisions and perform actions capable of modifying the environment and oriented to the achievement of the goals. For example, an air conditioner is turned on at the right temperature to solve a goal.

The result of composing the virtual profiles of the entities involved is not only the combined information of all the entities. It contains the combined history of the entities arranged in a single timeline, the result of the high-level inferences made on the combined virtual profiles, the set of goals of the entities, and their skills. From the combined Situational-Context information, strategies for achieving the goals based on the present skills should be identified. These strategies will guide the prediction of the interactions that should arise from the context and whose application will solve the goals detected in the environment. Let's see this with an example:

Let's imagine a smart store equipped with different smart devices, such as lamps, an air conditioner, switches, and a stereo. These devices can establish relationships with each other depending on the different situations in the store. Depending on the situation, specific lighting can be set, a certain type of music can be played, the temperature can be adjusted, or an electronic device (skills) can be turned on or off. In addition, contextual information such as date, time, location, or any other information that can be captured from the environment must be taken into account. For example, the level of brightness can vary from one part of the store to another to provide a better perspective of the products, or the music playing can be adjusted depending on the people inside the store (goals). This is because each situation has assigned values or goals that must follow a strategy to achieve the state of the comfort of the people in the store. To achieve the detected goals, entities must identify specific situations to act accordingly. This identification allows the entities to be aware of the situation they are in and to be able to make the necessary relationships to achieve their comfort level, setting the lighting level, the temperature, the type of music, or any other element that leads to solving people's needs. Figure 3.1 illustrates this example, where the air conditioner set a temperature to 21,5 degrees when two customers are in the store:

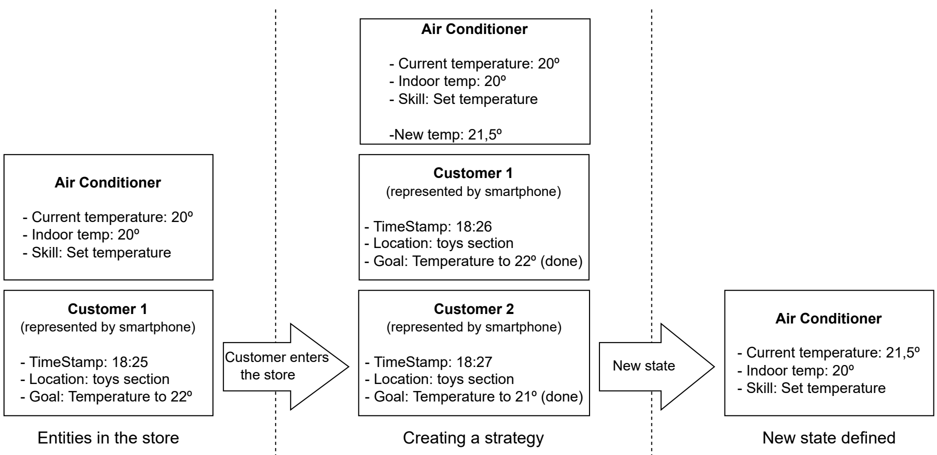


Figure 3.1: Situational-Context example definition

To support the situational context, a conceptual architecture has been developed. It is assumed that an intelligent social environment will possess entities consisting of IoT devices or people, which can give rise to different environmental situations. These entities have objectives that need to be agreed upon and resolved. These goals are the desired effects on the environment, such as setting a specific temperature, lighting level, or type of music. Entities also have skills that can be combined to resolve them. These abilities refer to actions that can cause a change in the environment at a given time, such as an air conditioner that can change the temperature, a lamp that can change the brightness or a stereo that can change the type of music. This architecture is intended to support the definition of all these elements, their integration, and coordination.

3.2 Architecture

The proposed conceptual architecture (Figure 3.2) contemplates that in intelligent scenarios, the different entities can communicate and exchange data to know the situation they are in and to be able to proactively perform actions to achieve common objectives. It has been designed to cover the different interoperability levels detailed previously in Section 2.2. Depending on the capabilities and tasks to be completed by the devices, two types of roles are considered in the architecture: entities and controllers. Entities are devices that can perceive and change the environment's state or have some needs or preferences. Controllers are devices that can request, receive and process information to identify what actions to execute to satisfy these needs. We note that a smart device can perform both functions with sufficient computing capabilities. In the following, we separate them for better readability.

On the one hand, entities E_n obtain values from the environment such as temperature or brightness (sensors) and perform actions to change their state (actuators). In addition to this information, entities have crucial information that must be shared for the correct management of interoperability, such as the identifier, brand, model, characteristics, goals/objectives, previous interactions, etc. To share this information, it is grouped in a wrapper that encapsulates what can be shared with different controllers when necessary. We consider that this wrapper is necessary because each entity has its own information that can be offered independently to other entities.

On the other hand, the controller is a device with sufficient computa-

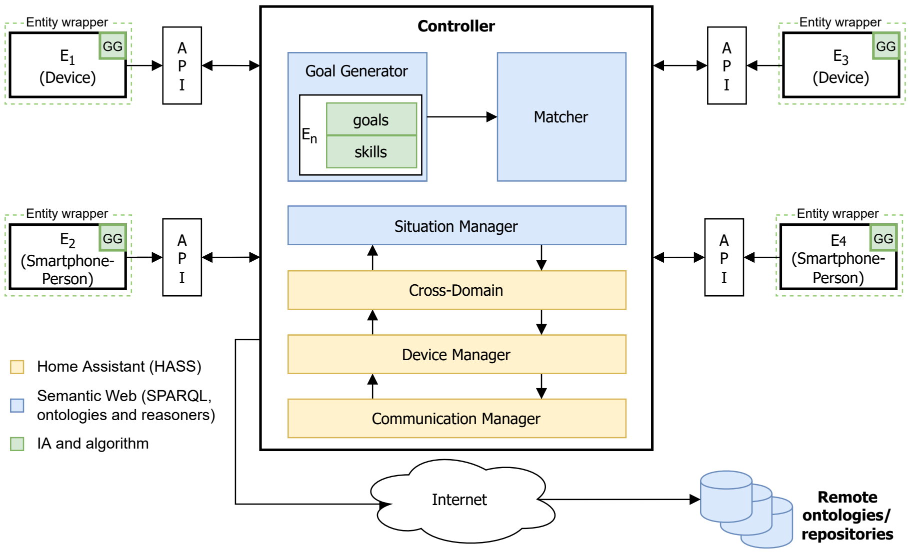


Figure 3.2: Situational-Context Architecture

tional capacity to process all the information obtained from the entities and decide which strategy (set of actions) should be activated based on the desired goals of the entities. Initially, the architecture assumes that the controller is physically located in the environment. However, the controller can also reside in the cloud or edge environments. To manage the interactions between the entities, the controller consists of different components:

- **Communication Manager**: it is responsible for achieving physical interoperability among entities. In this layer, multiplatform integration tools, such as Home Assistant (HASS) [1] or OpenHUB [2], allow connections at the network level, for example, to connect an entity *lamp* through WiFi, Bluetooth or Zigbee. This allows entities to be discovered and incorporated into the network environment to interpret later the information provided by each entity depending on the situation. Thus, the technological level is addressed. Entities can be discovered in the environment in several ways, through communica-

[1]https://www.home-assistant.io
[2]https://www.openhab.org

tion or network protocols. Once discovered, communication between them is enabled so that they can exchange information, which is discussed in the next component.

- **Device Manager**: the devices are managed to get to know the existing entities in the environment and their characteristics. This information will allow for determining the type of entity, which goals are pursued by each entity, and which skills are available to solve the needs of other entities. Each entity must be able to define its information in some way. Usually, each device has a typical document not written in machine language where its characteristics, functionalities, or configuration parameters are specified. This document is used as documentation or manual but not to improve interaction. It is static and can not be used to describe properties changing according to the context. The document can be provided by the manufacturer or generated following a specific specification. We will use the W3C Thing Description specification as detailed below. Therefore, this component is responsible for interpreting this document to give it a semantic connotation and make it easier to relate entities, regardless of the domain, as detailed in the following component.

- **Cross-Domain**: this component establishes relationships and provides dynamicity among different application domains. This requires the translation of data formats or languages in which device information can be specified to ensure data availability. For this purpose, content filtering and cleaning techniques can be used. So, it ensures these relationships hold regardless of the technology or language used in the domain to which an entity belongs. These relationships are achieved through the semantic connotation obtained from the information of each entity so that objectives and skills can be related as precisely as possible. The Semantic Web makes it possible to establish these relationships in a more or less simple way, as we will see below. Based on these relationships, the different situations that may occur in the environment are identified or created.

- **Situation Manager**: it identifies if a situation is already known in order to apply strategies that were successful in the past, or create new strategies. For this identification, contextual properties are used, such as the people present in the environment, the installed devices, or spatiotemporal data. This component uses the *communication manager* component to discover the devices, the *device manager* component to know how they interact, and the *cross-domain* component to know which ones can cooperate. This component is the most relevant component of the work and where the greatest efforts are applied and related directly to the situation level.

- **Goal Generator (GG)**: entities can have goals. They may be predefined by the manufacturer, manually defined by the user, or may be inferred by the presented architecture depending on previous interactions. This component is responsible for analyzing the previous interactions to discover the entities' goals. With the information provided about the situations, these goals can be generated. The goals can be of any type, such as turning on a lamp, increasing the temperature, setting a TV channel, recommending a certain product, etc. The generation of these goals facilitates their subsequent resolution. It is important to note that this component can be deployed in the controller, providing support to other low-powered entities, or can be deployed on the entities when they have enough computing capabilities, as shown in Figure 3.2.

- **Matcher**: once the objectives have been generated, this component identifies at run time which ones can be achieved using the skills offered by the different entities. For example, if a goal has been generated to increase the brightness of the room, and there is a lamp that can perform that function, these two entities are related to reaching that goal. This can be done by invoking the service of an entity that allows reaching the desired environment state. Thus, the different strategies to be carried out are established so that they can be carried out if the situation is detected again in the future. This must be done in real-time since the entities enter and leave the environments continuously.

These components can be implemented through different technologies. The following sections will detail the technologies selected for the implementation and provide the most important technical details to be considered. Let us describe an example to better define the interactions between these components. Imagine a lamp that can change the intensity and the color of its light. Thanks to the *Communication Manager*, the lamp can communicate with other entities, and the *Device Manager* is responsible for providing a global view of the environment. This view includes information about this particular device, such as its ID, communication interfaces, functionalities, etc. Furthermore, the user needs to integrate the lamp into a smart home where there are other smart devices such as an oximeter, to measure the amount of oxygen in the blood, and an audiometer, to measure the quality of hearing. Both devices belong to the healthcare domain, where data privacy and security are crucial. To provide a truly useful service, it would be interesting if the healthcare devices could send certain information to allow the lamp to change its light color depending on the results after a measurement. The *Cross-Domain* layer is responsible for enabling the exchange of information with other entities regardless of the domain. In addition, the lamp should also be able to change its intensity depending on the preferences of the people. The *Situation Manager* is in charge of identifying the situation that is occurring and determining if it should change the color or the intensity. The *Goal Generator* identifies if the people present have a lighting goal that must be satisfied. Finally, the *Matcher* finds and triggers the correct strategy.

3.3 Implementation

3.3.1 SMOTE: (S)ituation (M)anagement f(O)r Smar(T) (E)nvironments

SMOTE is an implementation of the proposed architecture for the management of situations in IoT environments. First, the entities according to the information provided in their wrappers are described. Second, the operations of the controller are specified, and its implementation is detailed. Finally, the dataflow of the process from the initial communication of the entities until the run-time selection of the services to adapt to the environment is shown.

3.3.1.1 Entity description

The information from the entities must be provided to the other entities and the controller must be analyzed and processed. This information is called entity description. The description of the entities follows the extension of the format proposed by the W3C, called Thing Description (W3C-TD) (Kaebisch et al., 2019). The W3C-TD is presented as a solution to counteract fragmentation in the Web of Things (WoT). It allows the developer to define smart devices using a standard based on the format JSON-LD (JavaScript Object Notation for Linked Data), detailing their id, title, security, or properties. However, the W3C-TD does not provide support for collaboration between entities (Flores-Martin, Berrocal, García-Alonso, & Murillo, 2020). Therefore, we propose to extend it with two sub-classes to support the definition of objectives and situations:

- **Objectives**: it details the objectives (goals) of the entity with regards to the environment. Each goal has its identifier, name, desired value, and contextual properties that define it, such as spatiotemporal data (time, location, etc.).

- **Situations**: it defines the situations in which the entity was involved. The information related to the situations is composed by an identifier, a name, a title, the contextual properties specifying the spatiotemporal data of the situation and the values of the solved goals (such as luminosity), and the strategies triggered depending on the detected goals. This information is later used to trigger the same strategies if they were successful, and to automatically infer new goals or changes in the ones already detailed.

The Figure 3.3 shows the extension of the W3C-TD class diagram where two new sub-classes are included to model these aspects (colored in green): *ObjectiveAffordance* and *SituationAffordance*.

These sub-classes follow the patterns of the main classes already defined which are *PropertyAffordance*, *ActionAffordance*, and *EventAffordance*. The two sub-classes allow two new types of interactions within the description of the *Thing* or entity, both to model its objectives and the situations in case it has them. It may be the case that an entity does

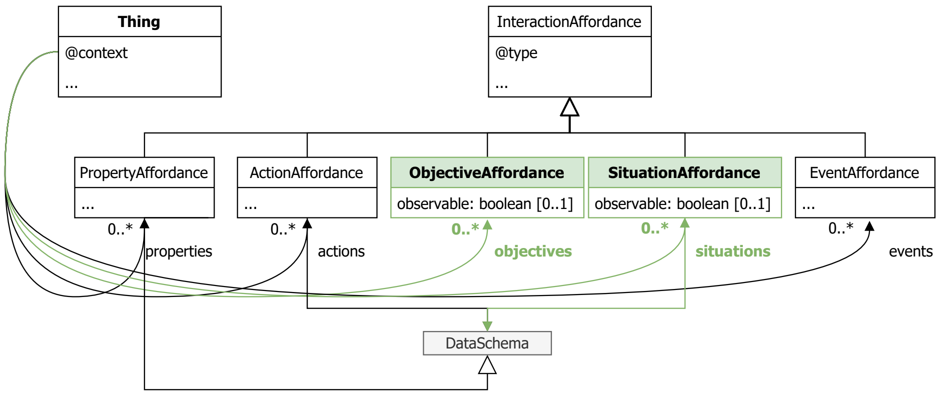


Figure 3.3: Extended class diagram from the W3C-TD

not have defined objectives, or that it does not have any situation stored. This leads to a description that only contains the requirements to promote collaboration between entities in social environments. This collaboration is later used to identify situations to promote proactivity in intelligent environments and to make their adaptation increasingly transparent to users.

Figure 3.4 shows an example of the description of a Yeelight smart light that will later be used in the case study. This description is a file specified according to the W3C-TD discussed above (blue). It shows the information of the entity such as the id or title, and the actions it has (such as turning it on or off, modifying the brightness, and customizing the color). The description also includes the two sub-classes specified above (red). On the one hand, the lamp has a goal to reduce its consumption. On the other hand, a situation is specified where the lamp has been previously involved in a party at home. This situation specifies contextual information (location, date, time, or luminosity), the objectives to be solved (luminosity), the entities involved (media players and other actors), and the strategy conducted (to increase the luminosity).

In the case of people, the file with its description is stored on their mobile device and will be consulted and modified according to the person's interactions. In the case of IoT devices, the description file could be provided by the manufacturer directly. The interpretation of this information

```
{
  "@context":"https://www.w3.org/2019/wot/td/v1",
  "id":"urn:dev:ops:32473-WoTLamp-1234",
  "title":"MyLampThing",
  "securityDefinitions":{ ⊞ },
  "security":[ ⊞ ],
  "properties":{ ⊞ },
  "actions":{
    "toggle":{
      "forms":[
        {
          "href":"https://mylamp.example.com/toggle"
        }
      ]
    },
    "luminosity":{
      "forms":[
        {
          "href":"https://mylamp.example.com/luminosity"
        }
      ]
    },
    "color":{
      "forms":[
        {
          "href":"https://mylamp.example.com/color"
        }
      ]
    }
  },
  "objectives":{
    "consumption":{
      "id":"reduce-consumption",
      "name":"Home consumption",
      "value":"30%",
      "properties":{
        "location":"39.4570601,-6.3835215",
        "time":"20:24:57 CET"
      }
    }
  },
  "situations":{
    "party-home":{
      "id":"party-home",
      "name":"Party at home for colleges",
      "properties":{
        "location":"39.4570601,-6.3835215",
        "date":"2020-04-24",
        "time":"21:30:00",
        "luminosity":"7",
        "weather":"sunny"
      },
      "objectives":{
        "type":"array",
        "items":[
          {
            "thing":"id-Bob",
            "objective":"id-luminosity",
            "value":"60%"
          },
          { ⊞ }
        ]
      },
      "things":{
        "type":"array",
        "items":[
          "id-Bob",
          "id-MediaPlayer",
          "id-Sara",
          "urn:dev:ops:32473-WoTLamp-1234"
        ]
      },
      "strategy":{
        "type":"array",
        "items":[
          {
            "thing":"urn:dev:ops:32473-WoTLamp-1234",
            "action":"luminosity",
            "value":"70%"
          },
          { ⊞ }
        ]
      }
    }
  },
  "events":{ ⊞ }
}
```


Original properties from the W3C Thing Description

Added extension to suppoort Objectives and Situations

Figure 3.4: Yeelight bulb description with the W3C-TD extension

is done through different techniques discussed below to detect the objectives and how they can be achieved. Therefore, each entity is responsible for creating and storing its own description. However, there may be entities that, due to hardware or cost limitations, can not store them. In these cases, it can be stored in remote repositories or in the controller of the environment.

3.3.1.2 Controller implementation

The controller is used to request and process the descriptions coming from the entities, to manage the interaction among entities, and to detect situations and trigger actions associated to them. In what follows, we present the implementation of the controller according to the components specified above and provide the key technical details. The pseudo-code for the whole process can be seen in Algorithm 1 while its notation is described in Table 3.1.

- **Communication Manager, Device Manager, and Cross Domain**: it has been decided to use an existing framework. The Home Assistant is the software in charge of managing the integration of the entities and addressing the three first levels of interoperability. We use this platform because it can support the first layers with minor modifications, allowing us to focus on the other layers. HASS supports a large amount of brands [3], and allows the integration of devices (sensors, actuators, mobile phones, etc.) through a large number of different protocols like Bluetooth, WiFi, or ZigBee. Through the API (Application Programming Interface) provided by each entity, HASS can establish communications and request their descriptions (*INIT()*). These descriptions allow HASS to identify the type of the entity and get its extended W3C-TD description, with information related to the situations and objectives stored in the entity. Also, some modifications were performed to enable automatic situation discovery when a new entity is detected in the environment. This has been achieved by requesting and processing the description of an entity when it is detected in the environment (*PARSEPDESCRIPTION(D)*). This is the basis for detecting and managing devices required by the following components.

- **Situation Manager**: a module has been implemented (*CHECKSITUATIONS(O)*) that is able to identify situations by taking into account the following parameters: the entities present in the environment, the objectives to be achieved, and spatio-temporal factors (location, date, and time). In this way, a situation is identified as

[3] https://www.home-assistant.io/integrations

known if these parameters are met, and its associated strategy would be launched. Otherwise, it would be a new situation and the required process would be conducted to elaborate the strategy and trigger it. These parameters are adjustable depending on the application.

- **Goal Generator**: from the -*skills*- that the entities have, as well as the goals of each situation, this component is in charge of identifying goals of each entity with the environment. This is done based on an entity's interactions with the environment and providing them with a complimentary name to be easily linked. This is done using a system proposed by the authors in (Rojo, Flores-Martin, Garcia-Alonso, Murillo, & Berrocal, 2020) that allows us to predict interactions with IoT devices in a smart environment through a neural network. In general terms, the network has as many input neurons as features have the entities. The number of outputs offered by the network is equal to the number of different skills that exist on the devices. When new devices are added, the number of outputs also changes, and the network is redefined to infer new goals.

- **Matcher**: Semantic Web technologies are mainly used to develop this component (*MATCHOBJECTIVES(O,S)*). We use ontologies and SPARQL queries to analyze what actions and objectives are present in the environment (Rhayem, Mhiri, & Gargouri, 2020). Currently, actions and objectives are linked by name. Each action and objective has a prefix, "sk_" and "g_" respectively, with which they can be identified and related. For example: "sk_illumination" and "g_illumination". In addition, in the case of similar names, the semantic reasoner is in charge of trying to establish this relationship. For example, "sk_illumination" and "g_brightness". When an action associated with an objective is found, it is performed to achieve the objective. E.g. "The room is a bit dark, I need to increase the brightness to level 7" (objective - "g_illumination"), by setting the light bulb to that brightness level (action - "sk_illumination"). With these pieces of information, we can identify situations. If the situation is known, it is enough to invoke the services associated with the strategy (*TRIGGERSTRATEGY(S)*). If not, a new situation is generated with the contextual information of the environment, the entities involved, and the associated strategy to be formulated. Then, this situation is sent to all the entities in the environment to be stored in their descriptions. The use of the Semantic Web can be combined with AI techniques (Seeliger, Pfaff, & Krcmar, 2019) to perform a different or more efficient matching according to the requirements of the application. In fact, AI is already used in the previous component for goal generation.

Table 3.1: Notation of the basic elements for the Algorithm 1

Element	**Description**
O	Defines the ontology where the entities will be stored. In this case: skeleton.owl.
E	Entities in the environment: devices or people (represented by their smartphones)
S	Situation that contains information about the environment. It is associated with an Entity
X(i)	Information. It could be referred to a different object (X) such as entities or situations
X(o)	Objectives (goals). It could be referred to a different objects (X) such as entities or situations
X(a)	Actions (skills). It could be referred to a different object (X) such as entities or situations
X(p)	Properties. It could be referred to a different object (X) such as entities or situations
X(s)	The different situations that an Entity or the Ontology posses
D	Corresponds to the Entity description written following the W3C-TD extension provided.

Currently, requests are stored in a queue to be processed on a first-come, first-served basis. In terms of complexity, it is worth mentioning that the system is easily scalable. The introduction of more controllers would make it possible to handle larger environments and more entities. In addition, more powerful controllers would provide shorter processing time and thus faster target resolution. The system could be extrapolated to larger environments, such as shopping malls, schools or museums. As future work we are defining new mechanisms to allow synchronization between different controllers.

Normally in each environment there will be one controller. However,

Algorithm 1 Environment processing

```
Function INIT():
    O ← loadOntology(skeleton.owl)
    name ← "w3ctde.daniel" ← detectedEntity
    D ← requestDescription(name)
    PARSEPDESCRIPTION(D)

Function PARSEPDESCRIPTION(D):
    E ← new Entity ;                                   // New thing
    E(i) ← D(i) ;                                      // Entity information
    E(o) ← D(o) ;                                      // Entity objectives
    E(a) ← D(a) ;                                      // Entity actions
    E(s) ← D(s) ;                                      // Entity situations
    file ← E.present(1) ;                              // Set entity as present
    ...
      O.add(E)
      CHECKSITUATIONS(O)

Function CHECKSITUATIONS(O):
    S ← O(s)
    for all S do
        if S(p) = True then
            properties ← True
        end
        if S(e) in O then
            entities ← True
        end
        ...
        if properties = True and entities = True then
            TRIGGERSTRATEGY(S)
            situation ← True
        end
    end
    if situation = False then
        MATCHOBJECTIVES(O,S)

    end
Function TRIGGERSTRATEGY(S,):
    actions ← S(A)
    for all actions do
        call(action(i), action(value))
    end
Function MATCHOBJECTIVES(O,E):
    S ← new Situation
    S(p) ← new Properties
    objectives ← E(o)
    action ← E(a)
    for all objectives do
        S(o).add(objectives(i)) action ← O.search(iri = objectives(i).name, class = Action)
        if action then
            call(action, objectives(i).(value))
            S(e).add(action(e))
        end
    end
    entities ← O(e)
    for all entities do
        if entities(i) in file = 1 then
            sendSituation(entities(i),S)
        end
    end
```

in larger environments, there may be more than one controller. Although there can be synchronization between controllers at the level of knowing which entities are connected to each other, what situations are occurring, or what objectives are being met, each controller must manage the entities that are close to it. This is so because if, for example, an objective to increase lighting is detected, the ideal is to select the closest controller to manage this request and select an entity that is also nearby to modify the lighting.

SMOTE provides support for the integration of different types of entities and for the identification of situations, all in a transparent way for the users. The following section presents the integration of the different devices and components to identify situations and activate strategies to achieve the objectives.

3.3.1.3 Dataflow

The dataflow in the controller is as follows: 1) an entity is detected; 2) its description is requested; 3) it is processed; 4) situations are identified or created; 5) the objectives associated with the situation are attempted. This step-by-step process is detailed in Figure 3.5.

1. **Detect entity and description request**: by using a script written in Nodejs, HASS identifies when an entity enters the network. HASS has a feature called "*device tracker*" [4] that allows detecting when a device enters or leaves the network range. This functionality can be configured in different ways: through Bluetooth/BLE, so that it detects close entities (+/- 10 meters), WiFi, which detects entities within the same network or a router that reports which devices are connected to it; or through the NMAP protocol, which allows scanning the network for devices. In our case, we will use the Bluetooth/BLE protocol to scan for devices, but this may vary depending on the application or the needs of the environment. When an entity is discovered, it is subscribed to the controller topic by using the MQTT protocol, and its description is requested.

[4] `https://www.home-assistant.io/integrations/device_tracker`

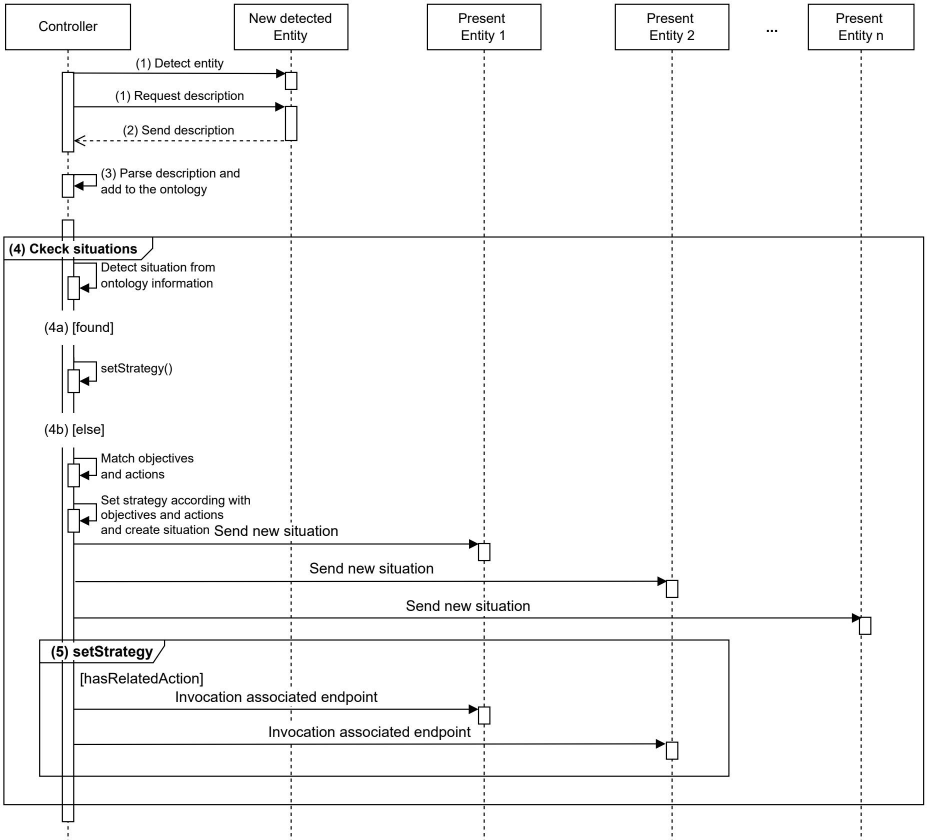


Figure 3.5: Sequence of the complete process in SMOTE

2. **Send description**: by using the MQTT protocol the entity publishes its description. In this way, the description is sent to the controller.

3. **Parse description**: the information of the entity is stored in an ontology based on the IoT-O ontology (Seydoux, Drira, Hernandez, & Monteil, 2016). This ontology contains the necessary classes that correspond to the description of the W3C-TD. All the treatment is performed by using Python, Owlready2 [5] and RDFLib [6] libraries,

[5]https://pypi.org/project/Owlready2
[6]https://github.com/RDFLib/rdflib

that provide us with methods and SPARQL queries to match skill and goals.

4. **Check Situations**: the *Situation Manager* detects the stored situations of the entity and interprets them to detect if any of the situations is taking place. At this point, two possibilities can arise: a known situation is detected (4a), or the situation is not detected and, therefore, is new (4b).

    - **4a) Known situation**: the strategies triggered in the past are identified and triggered again. The entities associated with the strategy are identified and their skills are invoked through the endpoints. These invocations include the associated value to set the action (e.g., luminosity - 7).
    - **4b) New situation**: the goals and skills of all the entities involved in a new situation are detailed in their description. The goals are identified using the *Goal Generator* component, or manually set by the user. When an unknown situation is identified, all this information is processed and the goals of the devices are covered with the available skills. The methods used in the *Matcher* component to make ontology queries are: in Owlready2, the *.search(...)* method of the ontology is used to get the *iri* (Internationalized Resource Identifiers) of the individuals, and in RDFLib the SPARQL method *.query(...)* is used to obtain information from the ontology at run-time. These two libraries are used in a complementary way to set a link. In the case of conflicting goals, such as different illumination levels for two people, the goal that is reached last will be considered. Finally, the situation is sent to all the entities in the environment to be added to their description.

5. **Set strategy**: once the strategy is identified, the endpoints of the entities are invoked. This is done by accessing the skill of a related entity that can do a change in the environment to achieve the goals.

This dataflow represents the basic operation of SMOTE, and can be applied to any application that meets the requirements of the architecture.

The basic aspects for the identification of situations, and how the devices should act according to them, have been controlled.

In this section, the architecture designed for the Situational-Context has been presented and an implementation of it has been provided. The validation of the architecture will be shown in Section 5. In the following section, we discuss the learning model followed by the Situational-Context and some of the paradigms that we considered for its development.

Chapter 4

Situational-Context: Learning Model

"Intelligence is the ability to adapt to change."
Stephen Hawking

Contents

One of the fundamental aspects of the Situational-Context is its learning model. This learning model allows situations to be identified in real-time based on environmental conditions and people's needs.

Different paradigms have been partially developed and used to achieve this behavior. Firstly, the **Human Data Model** paradigm, which allows defining users and devices data in a way that third-party applications can easily use; the **Human-Based Microservices Architecture**, which allows defining people-centered applications following a series of recommendations and best practices; and **Federated Learning**, a branch of machine learning aimed at distributed learning among several devices for decision making. This section will explain these paradigms and how they have been used during the development of Situational-Context.

4.1 Human Data Model

The "**Human Data Model**" (HDM), developed by Mäkitalo et al. (Mäkitalo et al., 2020) is a programming model for implementing Internet of Bodies (IoB) solutions, which is defined as a network of human bodies whose integrity and functionality depend, at least in part, on the Internet and related technologies (Matwyshyn, 2019).

Devices and users have different needs. As a result, not all devices can sense and process all types of data. For instance, today's most common IoB devices can be difficult (or impossible) to program. Instead, these devices are often connected to mobile devices or services and provide APIs to access the data. While many approaches aim to homogenize and provide APIs to the data collected with IoB, the problem remains that this information can be difficult to use in practice. One of the critical challenges of wearable technology seems to be usability and user experience. Imagine how difficult it could be for an elderly person to start interacting with a smartwatch, for this reason, an essential quality of HDM is that it can enhance interaction with various IoB and other types of devices, such as on a cell phone, cloud service, or even on a website and provide access to up-to-date data for user-friendly applications.

In this sense, HDM aims to merge the physical, cyber, and social worlds together, providing developers with an intuitive way to leverage data collected by services and devices while implementing new IoT applications. This is appreciated in Figure 4.1. In addition, the authors also developed an API to access and interact with it. It is an abstract model for collecting data relating to our lives and processing the data in an actionable form. With this model, it is also possible to build applications that proactively program computer-human interactions so that the computing infrastructure takes the initiative to better serve users. HDM documentation and implementation details are available in (Mäkitalo, 2019).

The number of interconnected devices and the amount of personal data they collect is increasing enormously. This makes it necessary to develop new tools to harness their potential. New devices are continually being introduced into people's daily lives and are already producing a wealth of data related to people's well-being. However, harnessing that information to create innovative Internet of Bodies solutions relies heavily on manually

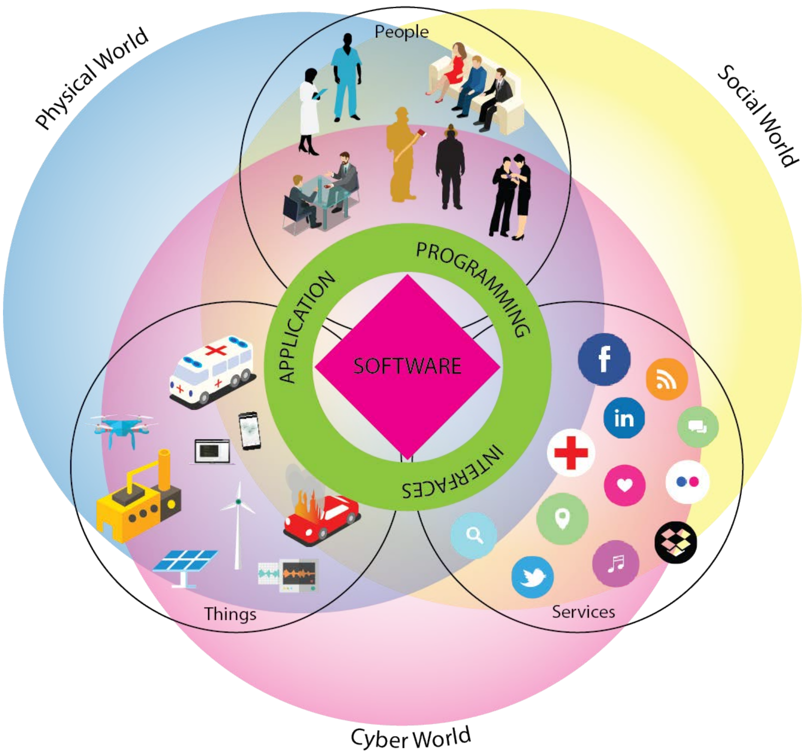


Figure 4.1: Unifying Physical-Cyber-Social Worlds

gathering the necessary information from various sources about the services and devices involved. With HDM, it is possible to combine personal information from various sources, perform calculations on that information, and proactively schedule computer-human interactions. Developers using the proposed model will have the opportunity to create solutions for the Internet of Bodies using high-level abstractions of users' personal information and taking advantage of the model's distributed approach.

The layered architecture and components of the Human Data Model are detailed in Figure 4.2. The *Programming Interface* layer executes the programming components of the Human Data Model. The *Human Data Model Instance Manager* (HDMIM) layer, on the other hand, is responsible for executing the programming constructs (*Analysis Model* component),

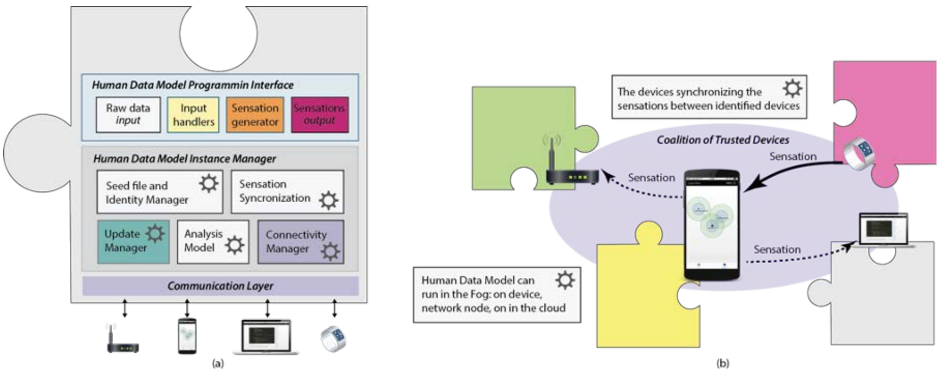


Figure 4.2: Human Data Model architecture

as well as connecting to other Human Data Model instances (*Connectivity Manager* component) and synchronizing data between them (*Sensation Synchronization* component) through the *Connectivity layer*. HDMIM also obtains seed files from the core of the Human Data Model. The seed files contain identifiers related to the person and their devices and services. With programming constructs, the framework merges the virtual, physical, and social worlds together, processing the data that these different worlds provide. The result-Sensations can be leveraged across all three worlds to create IoT applications. The API specification and some examples are available in the developer's repository (Mäkitalo, 2019). Thus, with HDM, abstract sensations can be generated that can be used in any application. The model is based on collective HDM executions, which simplify the development of applications that run in the cloud, in the fog, or at the edge, interchangeably.

HDM represents an important point of view concerning Situational-Context. First, it considers the use of heterogeneous devices to create more varied ecosystems, and, second, it unifies the data coming from these devices so that they can be used by third-party applications. In addition, its development mainly in Node.js served as inspiration for some of the examples carried out with Situational-Context. Devices and users possess different needs. As a result, not all devices can sense and process all types of data. For instance, today's most common IoB devices can be difficult (or impossible) to program. Instead, these devices are often connected to mobile devices or services and provide APIs to access the data. While many approaches aim to homogenize and provide APIs to the data collected with IoB, the problem remains that this information can be difficult to use in practice. One of the critical challenges of wearable technology seems to be usability and user experience. Imagine how difficult it could be for an elderly person to start interacting with a smartwatch. For this reason, an essential quality of HDM is that it can enhance interaction with various IoB types of devices, such as on a smartphone, on the cloud, or on a website, and provide access to up-to-date data for user-friendly applications.

4.2 Human-Based Microservices Architecture

We have seen how HDM helps to represent people's data in a simple and unified way to be used by IoT devices and applications. The next step is to analyze how these applications should be created to make it easier for developers and optimize the time and effort spent.

Therefore, another of the tasks developed for the Situational-Context was the development of an architectural pattern called "**Human-Based Microservices Architecture**" (HBMA). The purpose of this pattern is to offer developers a series of good practices where the development of human-oriented applications follows a series of guidelines. The HBMA pattern integrates people into the infrastructure of IoT applications and systems to interact with an intelligent environment. Therefore, it is considered useful to provide a guide for developers to perform this type of development in systems where it could be required.

To develop this pattern, different works were analyzed where it is clear that people are becoming more and more the core of software development (Miranda et al., 2015; Conti, Passarella, & Das, 2017; Guillen et al., 2013; Laso et al., 2021; Chen et al., 2017; Feng, Setoodeh, & Haykin, 2017). From these works, different aspects revolving around these developments were extracted, which were considered to develop the pattern, and which should be taken into account when developing people-based applications. These aspects are:

1. **A1. Personal communication interface**: How is the information transmitted? What device can be used? Latest developments

indicate the rise of the smartphone as the most widely used and comprehensive interface for interacting with other devices and collecting information.

2. **A2. Encapsulation and description of people's information**: What information should be provided? It is necessary to describe people's information so that they can be identified and distinguished from other people to conduct the actions required by the application.

3. **A3. Specification of personal information**: How is the information about individuals specified? In addition to being described, personal information must be specified to be easily interpretable and reusable by other systems.

4. **A4. Enrichment of information from different sources**: What type of information should be specified? Personal information can come from different sources that enrich the description of persons: personal data, social networks, interactions, etc.

5. **A5. Storing information about individuals on personal devices**: How and where should this information be stored? Although traditionally the use of the Cloud has been widespread for information storage, personal information should be stored on the person's own device for keeping privacy.

6. **A6. Providing personal information as a service**: What do we do with this information? The purpose of describing, specifying and storing this information is none other than to be offered to other people or devices in order to interact with smart environments and achieve more personalized services.

The next step was to develop the pattern. This pattern was designed following the recommendations described by Schmidt et al. (Schmidt, Stal, Rohnert, & Buschmann, 2013). These recommendations are based on specifying a series of minimum elements for the pattern: name, problem to be solved, a context where to be used, strengths, limitations, proposed solution, practical examples, example implementation, and relationship with other architectural patterns. Some of these elements are detailed below:

Problem: over time, people have become central to the development of software applications and systems. Information from people is useful for configuring IoT devices, adapting the conditions of an environment or recommending services. The traditional conception is that smart devices are the ones offering different types of services, while people are the ones consuming them. The current IoT approach is that people are also capable of offering services and that these are consumed by other people or devices. To do this, mechanisms are needed to deliver this information as a service within an intelligent environment. Some of the most important aspects of delivering this information are data privacy, data freshness, and information management.

Context: devices have traditionally been considered the core elements of IoT environments. This has brought great benefits and technological advances in the IoT field. However, the use of people's information to deliver personalized and tailored content, and also the trend to increasingly automate tasks, takes us to the next level where people increasingly need to provide their personal information. This is supported by numerous research papers providing paradigms and concrete implementations where the focus is no longer so much on devices, but on people.

Strengths and weaknesses: the objective of the pattern is to favor the development of applications and systems where people's information must be shared with the environment to favor interactions. This has several advantages:

- The main advantage lies in the **encapsulation of information** in a virtual profile, stored locally by a device linked to the person, such as a smartphone or tablet, and structured following a standard and easily interpretable format.

- Another advantage is to be able to establish different **levels of privacy** on this information and to be able to filter with whom this information is shared. In this way, the individual has control over his information, it does not travel to external servers, and he decides to whom he offers it.

- Another point in favor of the pattern is that **services are defined** to provide this information. Thus, third parties or devices in the environment can request it and act accordingly.

- These **services can be scaled** depending on the information established in the profile and the preferences of the person, who decides what to share and with whom.

Consequently, the main limitations have also been identified:

- Although almost everyone today has at least one smartphone, there may be instances where a **person does not have a mobile device** on which to store their profile and interact with the environment.

- **The amount of information** stored in the profile. Too little information would not be of great use, but too much information could saturate or slow down systems when querying and providing services. In this sense, it is also a limitation if it contains sensitive information about the person, as they may not want to share it or require additional privacy measures.

- **Network connectivity**, while considered more of a requirement, can also be a limitation. In smart environments, connectivity such as 4/5G, Bluetooth, or WiFi is necessary to interact with other devices in the environment. Having these services enabled at all times can lead to significant power consumption.

- **Shared devices**. A person often uses multiple devices, such as a smartphone, tablet, smartwatch, etc. This can lead to profile fragmentation or inconsistency of information, having to, at some point, be unified to ensure consistency of information. In addition, a device may be used by different people, such as a tablet shared with children, so be aware of who is using a device to associate this information with their profile and not someone else's.

- **Distributed system**. A distributed architecture always implies a higher level of complexity when designing, configuring, or maintaining the system. Therefore, both the design and management of an application that adopts such a pattern will require greater effort for the system to be truly efficient.

Solution: once the main information of the pattern has been defined, its composition is shown in Figure 4.3. The HBMA pattern consists of the following elements:

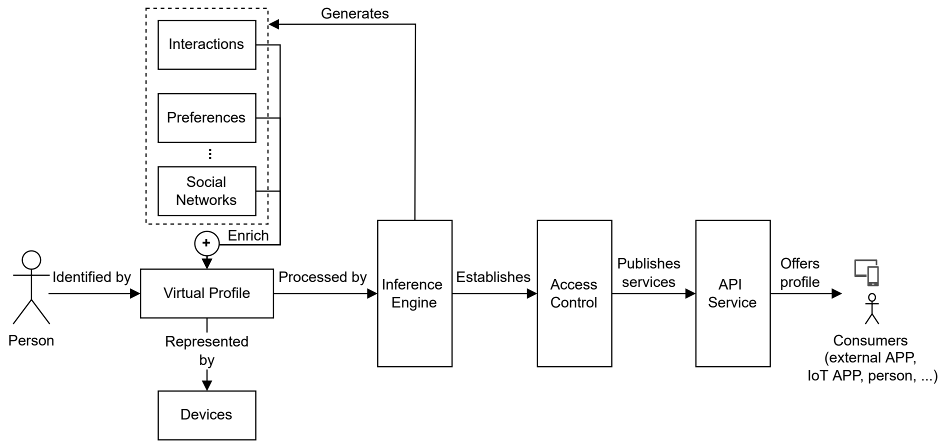


Figure 4.3: Human-Based Microservices Architecture pattern structure

- **Person:** is a natural person who is in an intelligent environment and wishes to interact with other people and devices using his or her personal information.

- **Virtual profile:** this profile can contain different types of information depending on the application to be developed. The most common information is the name, the age, the city, etc. One of the main aspects is that this information is enriched with information from other sources, such as the history of interactions with other people and devices, tastes and preferences, and social networks.

- **Devices:** the person is represented by one or more devices that act as a communication interface with the rest of the people and devices in the environment. This enables communication and information exchange. The most common device that has emerged as a reference representation of a person is the smartphone. In addition, the personal device(s) is/are responsible for storing the virtual profile locally.

- **Inference Engine:** the inference engine can generate the virtual profile from the interactions that the person has with his/her environment. These interactions are based on the exchange of information with other people or devices in the environment and taking into account the contextual information that surrounds them, such as spatiotemporal data, weather, the influx of people, etc.

- **Access control:** each person is the owner of his or her information and decides with whom and how to share it. It is necessary to decide with whom the information is shared, whether with the whole world, with elements close to the environment, or with elements already known beforehand. Privacy levels can be set according to the needs of the application to be developed: public, private, mixed, personalized, etc.

- **API Service:** the information must be provided to other people and devices in the environment. This filter is intended to establish which services will be available for consumption. This will vary depending on the information in the virtual profile, the degree of privacy applied, and the type of information contained in the profile.

- **Consumers (external APP, IoT APP, person, etc.):** third-party applications or people in the environment can query the person's virtual profile. This is done by consuming previously published services.

It is a fact that people-oriented development is becoming increasingly important. Because of this, the Situational-Context pursues just that, to be aware for adapting the environment to the people. The main problem arises because there are no standard practices or guidelines for developers to follow in order to design and implement these systems. The use of an architectural pattern is beneficial throughout the application development lifecycle.

This has its limitations. First, the use of an architectural pattern is limited to areas where its requirements are met. These areas are those in which people's information is primarily collected and used to generate and deliver personalized services. Developers considering the use of the pattern should consider some type of personal device as a communication interface with the environment and other devices. Special attention should also be paid to how information about individuals is described, as well as how and where it is stored. It is also important that this information can be enriched in different ways, such as through social networks or interactions with other people. And finally, it should be considered to offer all this information as different services are always under the control of the owner, who decides what to share and with whom. Considering these aspects, the application of the pattern would be satisfactory for the development of a people-based application.

That is why the HBMA pattern has been used to develop the Situational-Context, taking into account the different aspects that should be considered to create people-based applications. The next step was to provide the necessary intelligence to recognize situations efficiently. This was achieved by implementing a proprietary algorithm and taking advantage of Federated Learning techniques.

4.3 Federated Learning

Federated learning (FL) provides a training method to build personalized models without violating user privacy. The main feature of FL is to ensure user privacy by exchanging encrypted processed parameters. Its operation is based on storing data in different working nodes in a distributed manner and allocating resources through a reliable central server to efficiently obtain the final training model. In FL, compared to distributed machine learning, each worker node is the sole owner of its own data and a training participant of the model (Zhang et al., 2021).

One of the tasks within the development of the Situational-Context was the use of FL to develop a solution in which mobile devices play an active role (Rentero-Trejo, Flores-Martín, García-Alonso, Galán-Jiménez, & Murillo, 2021). Thus, the devices are able to learn the preferences of their owners and adapt the behavior of known and new intelligent envi-

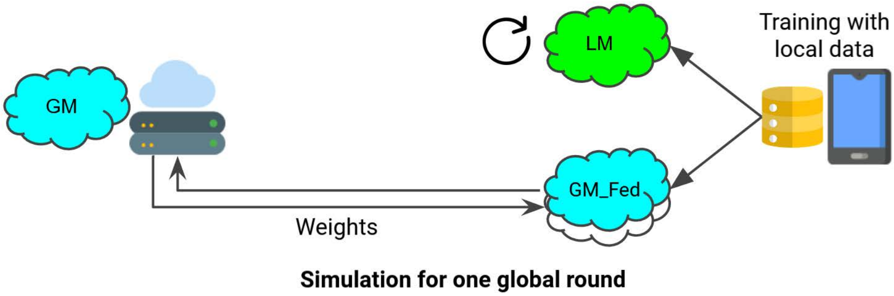


Figure 4.4: FL with a double local model architecture

ronments to these preferences. To obtain personalized and context-aware predictions, this solution proposed the consideration of two models. First, a global model with the knowledge generated in the federation and providing predictions to new users and/or new environments. Second, a customized model that adjusts to the needs of a specific user for already known environments. Both models can be retrained to meet federation and user needs. This approach allows for rapid customization and deployment, provides predictions on each environment the user visits, and manages multiple environments.

The data model follows the premises established by the Situational-Context, which considers the environment's elements, the type of device, the available actions, and spatiotemporal factors, among others. This way, it is possible to obtain an enriched data model that serves as input for the algorithm developed with FL.

This solution is based on standard FL and *FedAvg*, adapted to use a dual model. As mentioned above, there are two models. The first one, a **global model** is responsible for performing all the usual FL tasks (GM in Figure 4.4). Its goal is to create a global model from the federated users' knowledge, which is then downloaded to the user's device and used as a global knowledge base. And the second, a **local model** with similar characteristics to the previous one (LM in Figure 4.4), created in the device itself, which is responsible for training only with the user's data and without interference from external models (as in FL) to achieve personalized behavior.

On the one hand, the **global model** is built from the knowledge of the entire user network. First, each user has an empty model that is initialized by a set of parameters defined in the server, which provides further server control over the federation. Second, the model is trained over several epochs and sent to the server. This process occurs with all connected devices, and the server aggregates all the packets to create the global model. And third, the user gets this global model to provide a starting point for predictions. This method allows devices to make predictions based on the overall federated behavior. It is especially useful in situations where new users join the federation.

On the other hand, the **local model** is isolated on the users' device, and therefore, this model will not be affected by actions performed in the federation. The model is trained with the device's local data in the same way as the downloaded global model is. The main difference with the global model is that this model does not share its weights with any server and remains independent of the federated environment. The objective of this model is to obtain the highest degree of customization possible from the use of the device. The main advantage of this model lies in situations where the user is in one of his usual environments to make accurate predictions tailored to his preferences.

Summarizing, the global model is used as a source of general knowledge to determine what actions the federation would take in a similar situation. Over time, the global model will be replaced by the personalized model, which is the one that best knows the user's particular behavior. Both models compete to provide the best predictions; if one of them cannot give a sufficiently accurate prediction, the prediction of the model with the highest reliability index for that particular situation will be chosen.

The validation of the architecture was performed in two phases. First, the approach was validated using synthetic data, in order to choose the best configuration of the model. And second, the architecture is validated with real users to test its performance in real scenarios, as well as its impact on users' mobile devices.

In phase 1, the synthetic data led to better results overall, especially in familiar environments. In new environments, the performance of the local model is low, as the model does not yet have sufficient knowledge of the new situation, so it is necessary to use the global model. The performance will increase with the following actions in this environment until it becomes a usual environment and the local model is the most used. In phase 2, the models were tested with real data, and it was observed how the accuracies are smoothed to a more realistic degree. In this case, the accuracy increased considerably. The reason for this increase is that in real environments users tend to use the same type of devices, as they feel more familiar with them. The full results are available in the original article (Rentero-Trejo et al., 2021).

From the development of this approach, we learned that federated learning can be used for situation identification with very promising results. In addition, it also provided us with extra knowledge about IoT smart environments and how they work. Smart environments can be very different and heterogeneous, which makes the development of a multi-purpose and effective model for any context a challenging task. Also, interoperability is crucial to enable easy management of any IoT device via a mobile device. Another fundamental aspect is that the operation of these environments also depends on the context conditions. Each environment is defined by the devices in it and how they are used, which helps the model in its task of identifying the environment. Therefore the more properties and contextual information analyzed, the more appropriate the behavior will be and the more similar environments can be identified.

In this chapter, we have reviewed the main paradigms considered to develop the Situational-Context learning model. The Human Data Model (HDM) has allowed us to identify how people interact with IoT devices and how this information can be modeled. The Human-Based Microservices Architecture" (HBMA) architectural pattern has helped us to design and implement the Situational-Context according to a series of aspects that must be considered to create people-based applications. And, finally, Federated Learning has provided us with the necessary background to provide intelligence to the Situational-Context and to be able to identify situations more accurately.

In the following section, the Situational-Context is validated through the implementation of a real use case and different simulations carried out in environments with different conditions.

Chapter 5

Verification and Validation

"The only way to do great work is to love what you do."
Steve Jobs

Contents

In this section, we evaluate the feasibility and efficiency of the proposed Situational-Context architecture by using the SMOTE implementation in a case study. For feasibility, we analyze whether the ideas presented can be implemented in a real environment. For efficiency, we evaluate whether the response times obtained are suitable for IoT environments, where a certain processing speed is required. These two aspects are first evaluated in a real environment to obtain real values, and then in a simulated one to analyze the behavior of the system with a larger number of entities.

5.1 Case Study: A Smart Office

A case study based on a smart office has been developed where there are several entities, from IoT devices that regulate the brightness and turn on or off electrical appliances, to people whose preferences must be resolved by these devices. The idea is that as employees arrive, the devices will automatically adapt. To this end, when an employee arrives, his or her description is processed to identify whether the situation that is occurring is known or new. In this way, situations and the strategies associated with them are generated to adapt the devices to the employee's preferences. The first few days, the situations will be new: new employees, new devices, new routines, etc. Over time, these situations that are created are stored in people's descriptions. This makes it easier to identify an already known situation in the following days and automatically trigger its strategy, instead of having to do all the calculations to create a new strategy each time. The smart office is composed of the following entities:

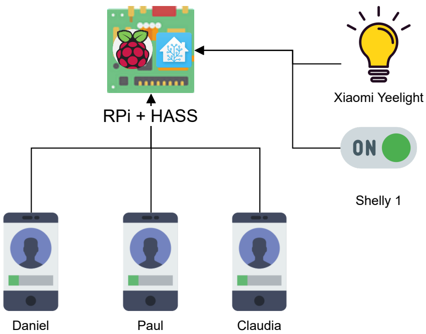
- **1 Raspberry Pi 4 Model B** (ARM Cortex-A72 Quad-core and 4GB RAM). A microcomputer takes the role of the controller, managing the different entities, and coordinating them to achieve the goals of a situation.
- **1 Yeelight v2 bulb** [1]. An IoT device with the skills of changing the color and the luminosity of a room.
- **1 Shelly v1** [2]. A switch with the skill of turning on or off the power supply of an appliance such as an A/C, a TV, a computer, etc.
- **3 Android Smartphones**: Belonging to three people and exposing their goals with the environment under different situations.
  - Honor 9: Kirin 960 Octa-core, 4GB RAM (*Daniel*).
  - Moto G6: Snapdragon 450 Octa-core, 4GB RAM (*Paul*).
  - Xiaomi Redmi 7: Snapdragon 632 Octa-core, 3GB RAM (*Claudia*).

We decided to use these devices, first, because they come from different IoT domains (smart home and industry), second, because they are quite common and their acquisition is easy and, third, because manufacturers provide an API that can be used to modify their behavior. In addition, with

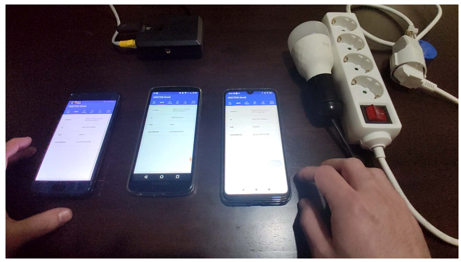
[1] https://us.yeelight.com/shop/yeelight-smart-led-light-bulb-1s-color

[2] https://shelly-api-docs.shelly.cloud/gen1

the information located in the descriptions of smartphones, we simulate different situations. For IoT devices and people, their descriptions are defined following the extension of the W3C-TD model. Figure 5.1 shows both the case study scheme (Figure 5.1a) and the real devices (Figure 5.1b).



(a) Case study scheme



(b) Case study real devices

Figure 5.1: Case study

This case study is based on the dynamic management of the entities' descriptions by the controller, following the whole process described in the proposed architecture. The entities are added and removed from the environment to check how the controller behaves and how the services of each entity are adapted to each situation. The smart bulb is activated or deactivated, regulating the level of brightness and even changing the color, and the switch turns on/off a socket strip. These behaviors vary according to the needs detected. For example, if a person enters the office and requires a certain level of brightness, the bulb sets the required brightness. Establishing one value or another depends on the current order of arrival for the entities, where the last entity to arrive will have priority over the others.

The case study allowed us to check the feasibility of the developed architecture by using SMOTE. The source code of the implementation of the controller and of the mobile application is available in public repositories [3], [4] as well as a complete video to explain the whole process of the

[3] https://bitbucket.org/spilab/server-node-python-w3ctde

[4] https://bitbucket.org/spilab/android-w3ctde

case study [5].

5.2 Performance tests

The performance tests are based on the case study developed in the Smart Office. This case study is focused on evaluating the behavior of the different entities and, especially, the controller. Depending on when the people enter or leave the office, the controller analyzes the goals of the people present to adapt the behavior of each smart device to the detected needs. The established value depends on the current order of arrival for the entities, where the last entity to arrive will have priority over the others. For this purpose, the processing time of the situations and the communication and execution time of the skills are evaluated. These tests were repeated 25 times under the same conditions, in order to obtain average values. The tests conducted were:

1. **To identify new situations (#1).** Given an environment with different entities, we treat the descriptions of these entities to discover new situations. This involves establishing a collaborative strategy and triggering actions to adapt to the environment.

2. **To process known situations (#2).** For an environment with different entities, we again treat the descriptions of the entities. In this case, since the situation has already occurred previously, we identify the situation that is occurring and automatically trigger the associated strategy to adapt to the environment.

3. **To measure the execution time for description parsing (#3).** A measurement of the average processing time of the entities in the environment is performed.

4. **To test the scalability of the system (#4).** Environments with more entities are simulated to test the feasibility of the system in larger environments.

[5] https://www.youtube.com/watch?v=zkRSt9d943o

For tests **#1** and **#2**, Figure 5.2 and Figure 5.3 show the times obtained for the management of the description for a new situation and for a known situation, respectively. The results obtained show that the processing time is considerably less when dealing with a known situation. Specifically, the average execution time obtained for new situations is 2,49 seconds, while for familiar situations it is 0,27 seconds. This is because in a known situation the associated strategy is triggered and it is not necessary to invest time on matching skills with goals to define the strategies. This means that as new situations are registered, their identification accelerates the adaptation of the devices to the environment.

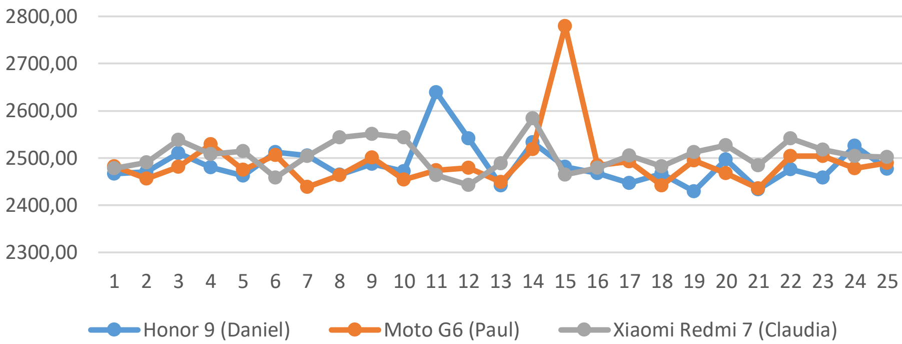


Figure 5.2: Description parsing for new situations (milliseconds)

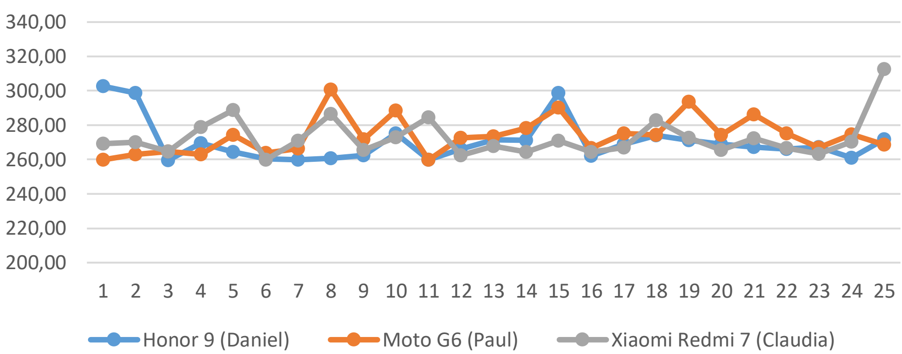


Figure 5.3: Description parsing for known situations (milliseconds)

Regarding test **#3**, the response time to process the entities' description has also been calculated. These tests consist of requesting and sending descriptions (MQTT) and triggering a strategy. The results obtained are shown in Figure 5.4. The times obtained vary considerably due to the functioning of the MQTT protocol. However, the average response time is 146,04 milliseconds for the three mobile devices, which can be considered acceptable taking into account the size of the description, the characteristics of the network, the number of devices and the experiments performed in (*HTTP vs. MQTT: A tale of two IoT protocols*, 2018).

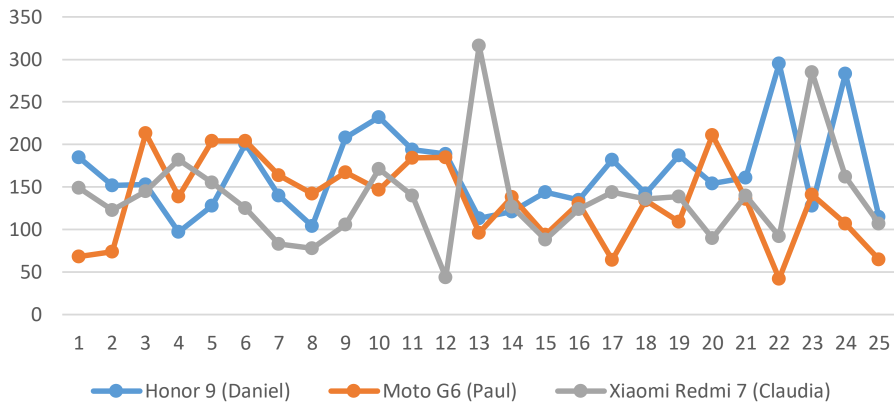


Figure 5.4: Time execution for description parsing (milliseconds)

The results obtained validate the feasibility of the system in a real environment. The response times for situation processing and the management of descriptions through MQTT are specific to the controller installed on a Raspberry Pi. These times are promising for this type of low-performance device and could be improved by using a more powerful dedicated server as a controller.

Also, test **#4** evaluates the scalability of the system, both at the network and information processing level, and it will depend on the type of controllers to be used. In our case, we used Raspberry Pi with up to 100 entities involved in the case study, and we calculated the processing times in the controller. Figure 5.5 shows the obtained times:

It can be observed that the average processing time per entity (descrip-

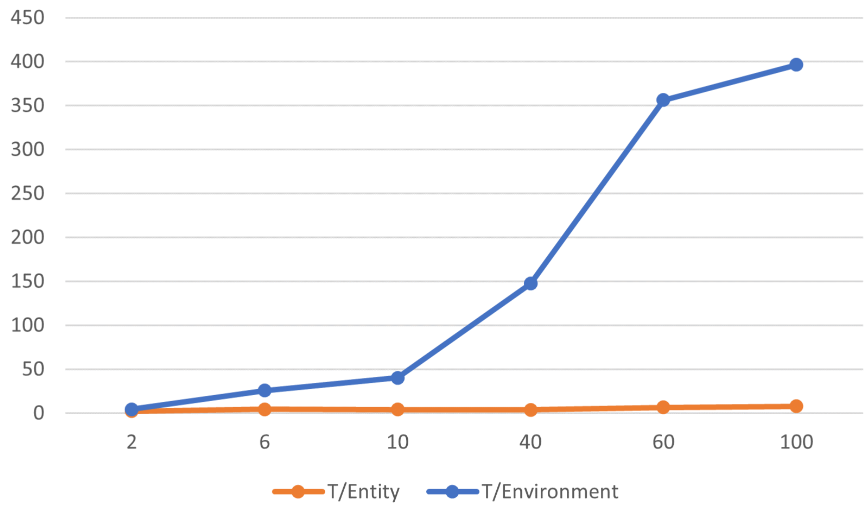


Figure 5.5: Average execution time for entities and environments (seconds)

tion processing) does not vary considerably with the environment (between 2,29 - 7,96 seconds). Hence, we conclude that even in very large environments this should not be a problem. Furthermore, the processing time for the complete environment, i.e., processing all descriptions, identifying the situations, and solving the objectives, does vary as a function of the size of the environment ranging over 4,58 - 396,31 seconds. Here it can be seen that in crowded environments the power of the controller should be higher.

5.3 Large environments simulations

To evaluate the feasibility and the behavior of the system in larger environments, seven environments were simulated where the number of devices and people vary from 1 to 50. To maintain consistency with the previous case study, the tests performed were repeated 25 times for each of the environments. These tests are focused on the following objectives:

1. **To resolve goals with the available entities (#5).** Evaluate how the detected objectives are solved with the available skills. To

do this, link the objectives and actions of entities that have common characteristics, e.g., the objective of increasing lighting with the action of turning on a light bulb, and create a strategy to follow. For example, lighting the lamp every day at the same time.

2. **To check the quality of the goals resolution (#6).** Evaluate how optimal is the link between objectives and actions. In this sense, false positives are detected, or in other words, objectives that were thought to be capable of being solved but could not be finally solved.

For tests **#5** and **#6**, the description of the entities contains 2 skills, 4 goals, and 1 situation. This configuration was selected because each IoT device has 2 skills that allow making changes in the environment, such as changing its state from on to off or changing the brightness; and people have 4 general goals, such as modifying the brightness of a room, setting a specific temperature, turning on a certain TV channel, or modifying the volume of the music. Therefore, a situation about working at the office can be discovered. To do this, up to 50 different descriptions corresponding to people and devices were generated. These descriptions contain skills and goals in a random way, simulating a real environment of arrival and leaving of people. The algorithm was modified to detect which of the predictions are accurate and which are not. This refers to finding an appropriate skill for a particular goal (*solvable goal*). For example, to increase the brightness of a room, the ability to increase the wattage of a light bulb in that room has to be identified. There may be cases where the skills and goals do not finally match, but due to the characteristics of the device or the similarity in skills and goals, the Matcher component identifies them as a match(i.e., it is a *false positive*). For example, for the goal of increasing the brightness, the ability to increase the energy savings of the bulb may be detected. False positives indicate that the goals have been identified as achievable but once the strategy was established they could not be achieved due to problems with the invocation of the associated services or because they were not fully supported. In the tests we considered different aspects to evaluate an environment: the number of devices and people, solvable goals (goals solvable with the available skills), solved goals (goals satisfactorily solved), and false positives (goals identified as solvable but not finally solved).

The results of these tests are detailed in Table 5.1 for each of the seven

Table 5.1: Solved goals and false positives after 25 executions per environment

Environment	Devices simulated	People simulated	Solvable goals	Solved goals	False positives
1	1	1	1,12	98,67%	17,33%
2	1	5	5,24	92,46%	16,80%
3	5	5	10,8	93,91%	33,06%
4	5	1	1,96	98,00%	33,33%
5	10	50	108,4	93,83%	33,49%
6	20	20	42,48	94,01%	34,67%
7	50	50	106,92	94,25%	34,34%

simulated environments. The descriptions vary to cover different possibilities that may occur in real environments. Knowing this, the system processes each environment to match actions and objectives and adapt the behavior of the devices. Two of the simulated environments are explained below.

In environment 1, 1,12 solvable goals are detected on average. As mentioned above, there may be other goals that can not be resolved because no associated skills are available. This applies to all simulated environments. The system was able to resolve 98,67% of these goals. In addition, 17,33% of the solved goals have been identified as false positives because they have been attempted to be solved with skills or devices that have not been able to perform it. For environment 5, 108,4 solvable goals are detected on average. In this case, the system has been able to resolve 93,83% of them, and 33,49% of the solvable goals have been detected as false positives.

Given the results, it can be deduced that the system promises great performance in crowded environments. The goal resolution has been satisfactory in 95,01% of the cases. In addition, the number of false positives is 29,00% for the simulated environments. The greater the number of devices and people in the environment, the more false positives will be obtained. This is because more objectives are detected to be achieved and, therefore, the probability of failing to achieve some of them also increases.

The evaluation performed above addresses the feasibility, efficiency, and performance of the architecture. We evaluated the latency and response times in tests **#1**, **#2**, **#3** and **#4**, and the performance and the quality experience of the "Situation Manager" and "Goal Generator" components in tests **#5** and **#6**, since these two components are the most important in the architecture and have been designed from scratch.

Chapter 6

Conclusions and Future Work

"The Internet? Is that thing still around?"
Homer Simpson

Contents

6.1 Conclusions

The development of this doctoral thesis has involved research in different areas, from the Internet of Things to Artificial Intelligence, including Semantic techniques and communication technologies.

In Chapter 1 the origin of the research line was established, and how as a result of open lines such as People as a Service and Internet of People, the idea of Situational-Context was born. In addition, the problem detected and the objectives to be met were detailed, as well as the contributions that were expected from this thesis and that have finally been achieved. The collaborations that arose both in research stays with other universities and in research projects were also described. These collaborations served to enrich the background on the subject and to know how other research groups work.

Once the problem to be solved had been established, an analysis of the state of the art was carried out. In Chapter 2 several key aspects of the development of the Situational-Context were analyzed. These aspects were the mechanisms of collaboration in IoT environments and the main pillars that refer to the related areas and that represented a potential contribution to the development. Once this analysis was completed, a Systematic Literature Review was performed where the works related to these pillars were analyzed. These works allowed us to detect the gap where the Situational-Context, and therefore this thesis, could make a real contribution. This gap was the identification of situations in collaborative IoT environments, and this is where we focused our development.

The results of the literature analysis provided us with the necessary background to start with the development. In Chapter 3 the concept of Situational-Context is introduced to provide the reader with a more detailed view of this paradigm. In addition, we start with the development by defining an architecture that supports the gap detected in the Systematic Literature Review and that aims to meet the objectives established at the beginning of the document. For this architecture, a series of modules are defined that allow from the physical communication of the devices to the identification of situations and subsequent automatic adaptation of the environments. In addition, the design of the entities is detailed, and how their information must be specified and treated to facilitate the work of the developers. It also details the implementation of the environment controller element, which is in charge of processing the information of the entities, identifying the situations, and establishing a strategy for them. Finally, the data flow of the whole process is described, from the moment an entity is detected in the environment, all its information is processed, and until the devices are adapted to its preferences. This allowed refining the operation of the architecture components to subsequently develop the learning model.

The learning to identify situations is given by different aspects. In Chapter 4 the methodologies used to develop the Situational-Context learning model were studied. These methodologies focus on defining a data model to organize people's information that can be used to offer different types of services. These services will be invoked by the environment controller to adapt the behavior of the devices. In addition, a federated learning model was designed, which, based on the knowledge generated by all the entities in the environment, makes it possible to establish a personalized learning mechanism for each entity. This allows the creation of simpler and lighter learning models. In this way, the controller can predict future situations for the entities to provide a first adaptation of the environment to the users.

With the development done, it was time for testing, what it was addressed in Chapter 5. Once the architecture and its learning model had been defined, a real case study based on a Smart Office was implemented to evaluate the efficiency, effectiveness, and feasibility of the architecture. Different tests were performed to check response times, and processing times and even larger environments were simulated to check the scalability of the system. Finally, these tests concluded that the architecture designed for the Situational-Context was perfectly viable to be taken to a production environment.

Given the work conducted, we can conclude that the objectives set out at the beginning of this doctoral thesis have been satisfactorily met, proposing a viable solution for the automatic adaptation of intelligent environments based on information from people.

6.2 Publications

6.2.1 Accepted papers

The scientific publications derived from this thesis are listed below in chronological order:

- **Flores-Martin, D.**, Berrocal, J., García-Alonso, J., & Murillo, J. M. (2022). Towards Dynamic and Heterogeneous Social IoT Environments. Computing. **(Q2, 2.420)**

- Rentero-Trejo, R., **Flores-Martín, D.**, Galán-Jiménez, J., García-Alonso, J., Murillo, J. M., & Berrocal, J. (2022). Using Federated Learning to Achieve Proactive Context-Aware IoT Environments. Journal of Web Engineering, 53-74. **(Q4, 0.617)**

- **Flores-Martin, D.**, Rojo, J., Moguel, E., Berrocal, J., & Murillo, J. M. (2021). Smart Nursing Homes: Self-Management Architecture Based on IoT and Machine Learning for Rural Areas. Wireless Communications and Mobile Computing, 2021. **(Q3, 1.819)**

- **Flores-Martin, D.**, García-Alonso, J., Berrocal, J., Foschini, L., & Rodríguez, J. M. M. (2021, September). Context-Dependent Services Selection in Smart Environments. In 2021 IEEE Symposium on Computers and Communications (ISCC) (pp. 1-6). IEEE. **(Class 3)**

- **Flores-Martin, D.**, Mäkitalo, N., Berrocal, J., García-Alonso, J., Mikkonen, T., & Murillo, J. M. (2021, April). Layered Interoperability for Collaborative IoT Applications. In Future of Information and Communication Conference (pp. 192-211). Springer, Cham.

- **Flores-Martin, D.**, Berrocal, J., García-Alonso, J., & Murillo, J. M. (2021). SMOTE: A Tool to Proactively Manage Situations in WoT Environments. In International Conference on Web Engineering (pp. 525-529). Springer, Cham. **(Class 3)**

- **Flores-Martin, D.**, Laso, S., Berrocal, J., & Murillo, J. M. (2020, October). Detecting and Monitoring Depression Symptoms According to People's Behaviour Through Mobile Devices. In International Workshop on Gerontechnology (pp. 3-10). Springer, Cham.

- Mäkitalo, N., **Flores-Martin, D.**, Berrocal, J., Garcia-Alonso, J., Ihantola, P., Ometov, A., ... & Mikkonen, T. (2020). The internet of bodies needs a human data model. IEEE Internet Computing, 24(5), 28-37. **(Q1, 4.231)**

- **Flores-Martin, D.**, Berrocal, J., García-Alonso, J., & Murillo, J. M. (2020, June). Extending W3C Thing Description to Provide Support for Interactions of Things in Real-Time. In International Conference on Web Engineering (pp. 30-41). Springer, Cham.

- Rojo, J., **Flores-Martin, D.**, Garcia-Alonso, J., Murillo, J. M., & Berrocal, J. (2020, March). Automating the interactions among IoT devices using neural networks. In 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops) (pp. 1-6). IEEE.

- **Flores-Martin, D.**, Pérez-Vereda, A., Berrocal, J., Canal, C., & Murillo, J. M. (2018, December). Interconnecting IoT devices to improve the QoL of elderly people. In International Workshop on Gerontechnology (pp. 83-93). Springer, Cham.

- **Flores-Martin, D.**, Canal-Velasco, J. C., Berrocal, J., & Murillo, J. M. (2019). Abordando los distintos niveles de colaboración entre dispositivos en entornos IoT.

- **Flores-Martin, D.**, Laso, S., Berrocal, J., Canal, C., & Murillo, J. M. (2019, September). Allowing IoT devices collaboration to help elderly in their daily lives. In International Workshop on Gerontechnology (pp. 111-122). Springer, Cham.

- Laso, S., **Flores, D.**, Garcia-Alonso, J., Murillo, J. M., & Berrocal, J. Deploying APIs: Edge vs Cloud Environments. MMTC Communications Frontiers.

- **Flores-Martín, D.**, Pérez-Vereda, A., Berrocal, J., Canal-Velasco, J. C., & Murillo, J. M. (2019). Coordinación de dispositivos IoT mediante web semántica y ontologías en situational-context.

- **Flores-Martin, D.**, Berrocal, J., García-Alonso, J., Canal, C., & Murillo, J. M. (2019). Enabling the interconnection of smart devices through semantic web techniques. In International Conference on Web Engineering (pp. 534-537). Springer, Cham. **(Class 3)**

- **Flores-Martin, D.**, Berrocal, J., Garcia-Alonso, J., & Murillo, J. M. (2019, March). Towards a runtime devices adaptation in a multi-device environment based on people's needs. In 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops) (pp. 304-309). IEEE.

- **Flores-Martin, D.**, Pérez-Vereda, A., Berrocal, J., Canal, C., & Murillo, J. M. (2018, December). Interconnecting IoT devices to improve the QoL of elderly people. In International Workshop on Gerontechnology (pp. 83-93). Springer, Cham.

- **Flores-Martin, D.** (2017, November). Meeting IoT users' preferences by emerging behavior at run-time. In International Conference on Service-Oriented Computing (pp. 333-338). Springer, Cham.

6.2.2 Pending papers

In addition, some papers are pending acceptance:

- **Flores-Martin, D.**, Berrocal, J., García-Alonso, J., Canal, C., & Murillo, Juan M. An Architectural Pattern for Providing People's Information from their Companion Devices. Journal of Systems and Software. **(Q2, 3.514)**

- **Flores-Martin, D.**, Rentero-Trejo, R., Galán-Jiménez, J., García-Alonso, J., Murillo, J. M., & Berrocal, J. Knowledge Sharing in Proactive WoT Multi-Environment Models. Journal of Web Engineering. **(Q4, 0.575)**

- Laso, S., **Flores-Martin, D.**, Herrera, J. Luis, Galán-Jiménez, J., & Berrocal, J. Identification and visualisation of a patient's medical record via mobile devices without an internet connection. Electronics. **(Q3, 2.690)**

6.3 Future Works

Future work derived from this thesis focuses on defining a hierarchical or prioritization mechanism for generating and establishing strategies for situations. Based on different properties of people, such as the frequency of going to a particular place, the time spent in that place, or the degree of affinity with the environment, we are working to ensure that the strategies that adapt to the situations take into account the preferences of certain people based on these properties.

Another line currently open is to improve the programming model with federated learning mechanisms. Within federated learning, we are applying knowledge distillation techniques. Knowledge distillation allows us to generate simpler learning models to predict situations. This is intended to reduce processing time when identifying situations and activating their strategies.

We are also working to define an architectural pattern to develop people-centered applications in a simpler and standard way. As a consequence of the Situational-Context, we detect that there is a growing need for applications to be more people-centered and not so much device-centered. This is why the development of an architectural pattern is necessary to provide a development guide to developers when implementing this type of application.

6.4 Final Reflection

The realization of this thesis began back in 2017 when I finished my Degree studies and started my Master's studies. At the same time that I was doing my Master's studies, I began to immerse myself in the research world and all that it entails. In the beginning, it was complicated to combine both tasks, but as time went by I got used to the research tasks, always guided by the good judgment of my directors.

During this period, I have been able to learn new areas of knowledge, technologies, techniques, and methodologies that I did not know and that have enriched my professional career. This has allowed me to develop my skills as a computer engineer while acquiring the necessary skills for a research career. In addition, collaborations with colleagues, other universities, and the research stays have given me an additional vision of seeing things that perhaps I did not have before.

The development of the thesis has also given me the opportunity to participate in different research projects, where I learned how a project works, how the tasks are managed, and the importance they have within a research career. In addition, they have provided me with additional knowledge and collaborations with other colleagues both from my university and from other universities, an aspect that I consider very positive and enriching. I have also had the opportunity to start with some teaching tasks.

First, through participation in different teaching innovation projects whose objective was to improve learning methodologies and increase student motivation. And secondly, I had the opportunity to teach in different subjects, which has allowed me to gradually acquire the necessary skills for a teaching career. These teaching tasks have given me a lot in both my professional and personal life, having to deal with many people and manage different aspects that require significant time management and coordination.

The development of this doctoral thesis has been an important personal challenge for me, with good and not-so-good moments. It is a complicated path full of uncertainty, but it has also brought me many good things in my life, and that is what I keep in the end. I have met many people who have helped me and with whom I have shared countless incredible moments. However, it is worth mentioning that the development of this doctoral thesis was involved in the global pandemic of COVID-19. This meant dealing with all that this entailed, and it was necessary to rethink the tasks and methodologies used during the development to adapt them to the conditions of the pandemic.

To conclude, I consider that this period has been very fruitful for me and that the work done has provided me with an important base as a researcher but has also made me grow and mature as a person. That is why I finish my doctoral thesis with new skills and reinforce the ones I already had to face any challenge that life throws at me.

"Show must go on."
Freddie Mercury

Appendix I

SLR Methodology

This appendix details the activities performed to develop the SLR in accordance with the (Kitchenham & Charters, 2007). Each of them is detailed below.

I.1 Research tools

Different tools were used to conduct the SLR phases. These are:

- **Mendeley** [1] as bibliographic manager. Mendeley is an advanced bibliographic manager that allows storing, ordering, and simply reading documents. This tool was used to read scientific papers and take notes on them.

- **Parsif.al** [2] as the main data extraction tool. Parsif.al is an online tool that, through a series of steps, allows to perform a complete SLR. This tool was used to import the documents from Mendeley and make a selection of those to be finally analyzed. In addition, Parsif.al allows the creation of forms to manage the quality of each paper based on the objectives of the defined SLR, as well as the extraction of research data and their analysis.

[1] https://www.mendeley.com
[2] https://parsif.al

- **Microsoft Excel** [3] as a complementary tool for managing the SLR. Parsif.al allows exporting the results to a *.csv* file. This file can be manipulated with Excel and was used to create customized statistics and reports.

I.2 Plan Review: Protocol planning

I.2.1 Background

As mentioned, this SLR is intended to provide the necessary background to develop the Situational-Context. For this purpose, work focused on providing situation-dependent interoperability in IoT smart environments and favoring its adaptation to people's preferences will be reviewed. To this end, three main needs (N) are identified:

- **N1**. Identify approaches based on IoT solutions.

- **N2**. Identify the IoT standards that allow improving interoperability between smart devices.

- **N3**. Identify which architectures, middlewares, frameworks, or systems are used in other works to adapt to situations in IoT environments.

The next step is to establish the research questions based on these needs.

I.2.2 Research Questions

The research questions are one of the most crucial parts of SLR development. They must be specified correctly since they condition the rest of the work. Therefore, it is necessary to establish the right questions to solve the detected needs. Table I.1 specifies the research questions posed for this SLR.

[3] `https://www.microsoft.com/es-es/microsoft-365/excel`

Table I.1: SLR Research Questions

#	Research Question	Motivation	Related to Need
RQ1	What technologies exist for managing IoT/WoT smart environments?	Know the existing applications or frameworks that allow connecting IoT devices to interact with the environment. There are platforms that allow installing devices that, through a previous configuration, can automate routines or make changes in the environment. For example, using a light bulb with Alexa.	N1
RQ2	What standards exist to provide interoperability between IoT devices?	Know the most popular protocols for exchanging information between devices. There are many communication and information exchange protocols on the market. This heterogeneity allows having a wide range of devices, but on the other hand it makes compatibility between devices difficult. For example, using the Semantic Web.	N2
RQ3	What architectures/middlewares exist to provide interoperability between IoT devices?	Know the techniques used in existing works that favor interoperability. The problem of heterogeneity in IoT has been addressed in many existing works. We aim to analyze these works to learn about the techniques employed and how they can be applied in the context of this thesis. For example, using Home Assistant or Open Hub.	N2
RQ4	What kind of interoperability do current solutions provide?	Know the types of interoperability addressed in each paper. There are many levels of interoperability and the current works tend to focus on providing solution to them individually. The aim is to analyze the works that address these levels individually or jointly and how they are treated in IoT environments. For example, interoperability at the level of connectivity and information exchange, or at the semantic level.	N2/N3
RQ5	What techniques are used to adapt devices to people?	Know what techniques are used for devices to adapt their behavior. There are many elements that affect the adaptation of an IoT environment, such as the context, nearby devices, the people involved, etc. The aim is to analyze how these elements are treated to decide the behavior of a device to a given event. For example, setting the brightness of a light bulb.	N3

I.3 Conduct Review: SLR execution

I.3.1 Search Criteria

Once the research questions were established, the next step is to establish the search criteria. To do this, the sources of scientific resources to be searched must be selected. Among the most famous sources are Web of Science (WOS) [4], Google Scholar [5], or Scopus [6]. For this SLR we have decided to use **Scopus** because it is one of the most complete and relevant sources for locating papers related to Computer Science.

I.3.2 Search String

The search string will depend on the terms we want to locate in the papers. In addition, alternative terms can be entered for each term to cover a wider range of possibilities. The terms used for this SLR are listed in Table I.2.

Table I.2: SLR Search string composition

Main Term	Alternative Terms	Related to Question
"Internet of Things"	IoT OR "Web of Things" OR WoT OR "Smart Environment"	RQ1
Interoperability	Interconnection OR Connectivity OR Communication OR Protocols	RQ2/RQ4
Context	Situation OR "Situation Aware" OR "Situation-Aware" OR "Context Aware" OR "Context-Aware"	RQ4
Architecture	Platform OR Middleware OR Framework OR Standard	RQ2/RQ3
Adaptation	"Self-adaptation" OR Behaviour OR Needs OR Preferences OR People	RQ5

As can be seen, papers are desired that discuss the Internet of Things, interoperability, context, architectures, and adaptation, each of these terms with their respective alternative terms. From these terms, we can establish the search string that will allow us to locate the jobs to be analyzed. The search string goes through several modifications until the final one is

[4] https://www.webofscience.com/wos/woscc/basic-search
[5] https://scholar.google.com/
[6] https://www.scopus.com

decided. In addition, a filter was used to limit the search area to computer science (COMP). Table I.3 shows the search string finally used.

Table I.3: SLR Final Search String

Search string
(TITLE-ABS-KEY ("**Internet of Things**" OR "IoT" OR "Web of Things" OR "WoT" OR "Smart environment") AND TITLE-ABS-KEY ("**Interoperability**" OR "Interconnection" OR "Connectivity" OR "Communication" OR "Protocols") AND TITLE-ABS-KEY ("**Context**" OR "Situation" OR "Situation aware" OR "Situation-aware" OR "Context aware" OR "Context-aware") AND TITLE-ABS-KEY ("**Architecture**" OR "Platform" OR "Middleware" OR "Framework" OR "Standard") AND TITLE-ABS-KEY ("**Adaptation**" OR "Self-adaptation" OR "Behaviour" OR "Needs" OR "Preferences" OR "People")) AND (LIMIT-TO (SUBJAREA , "COMP"))

I.3.3 Inclusion and Exclusion Criteria

The inclusion and exclusion criteria allow us to determine what type of work will be analyzed later. These criteria are intended to find technical and technological depth, based on scientific facts. These criteria are decided based on other SLRs that may be related to the same area and also based on the expert knowledge of other researchers.

On the one hand, the studies that meet the following conditions were **included**:

- Complete papers.
- They must belong to the "*Computer Science*" branch.
- Any type of publication, whether journal, conference, doctoral thesis, etc., give preference to indexed journal papers and conferences.
- Studies that present a method and/or technique to improve interoperability between devices in IoT/WoT environments.
- Publications until 2020[7].

[7] This SLR was conducted during the year 2020

On the other hand, studies that met at least one of the following conditions were **excluded**:

- Papers that do not specifically address technologies in IoT/WoT environments are not of interest to the study, or simply name IoT/WoT as an example without being the main focus of the study.
- PowerPoint or similar presentations.
- Duplicate papers or papers coming from the same source.

I.3.4 Data Extraction Strategy

The execution of the search string returned a total of **614** results, as can be seen in Figure I.1. A series of inclusion and exclusion criteria will be applied to these results to determine which papers will be analyzed.

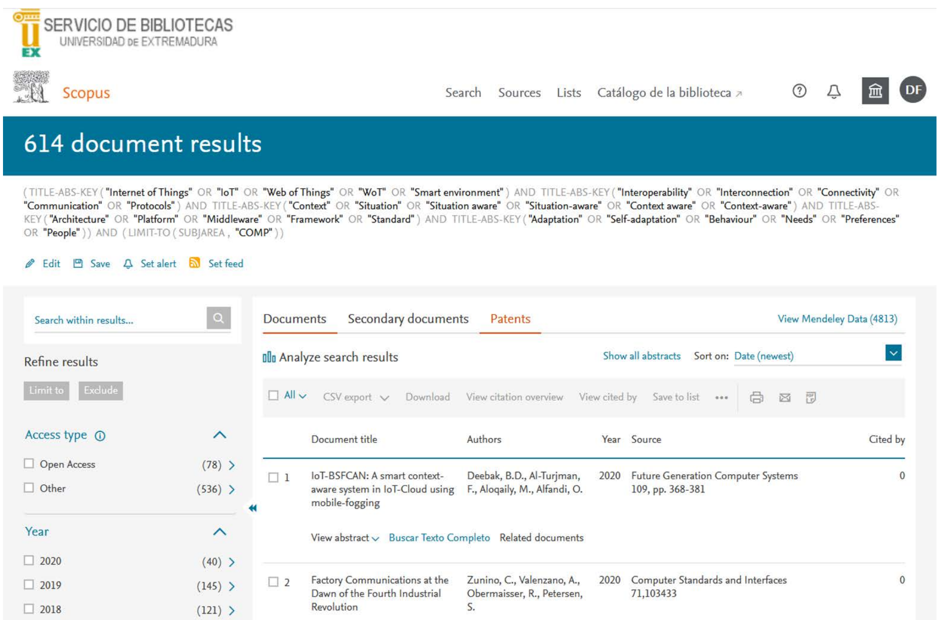


Figure I.1: SLR first search result

To this must be added papers that had previously been identified from other sources such as conferences and specific journals, and were considered relevant but did not enter the search string. In addition, these results were complemented with post-2020 papers discovered during the development of this thesis, so finally we processed **621** papers. This process is illustrated in Figure I.2 and detailed below through 4 main phases:

1. First, the search for all items was performed by attending to the search string established earlier. To these papers were added those that were previously known and were relevant to SLR, resulting in a total of **645** papers.

2. Second, duplicate papers were discarded, resulting in a total of **638** papers. In this phase, the analysis of the papers started, based on the title, abstract, and keywords.

3. Third, for each paper analyzed in the previous phase, the previously defined inclusion and exclusion criteria were applied. This makes it possible to determine which papers are finally fully analyzed. The result of this phase yielded a total of **53** papers to be analyzed by title, abstract, keywords, and full text. The main exclusion criterion was that many of the papers deal with IoT to create specific applications but do not address interoperability or they are not context-aware.

4. Fourth, the last filtering was performed to determine which of the analyzed papers would finally be included in the analysis. It was determined to all the papers since the IoT trajectory is not too great in terms of improving interoperability and we think that the papers analyzed more than meet the restrictions established in this SLR. However, we could not access one of these, so finally, **52** papers were included in the analysis.

Following the steps provided by the Parsif.al platform, the data extraction for the **52** papers were divided into two parts.

On the one hand, a checklist was made to quantify the suitability of each work analyzed concerning the SLR objectives. This checklist is based on a series of questions (Table I.4a) to which a score is assigned according to the content of each paper (Table I.4b). Some of these questions are based on whether architecture is properly described, whether people's needs are

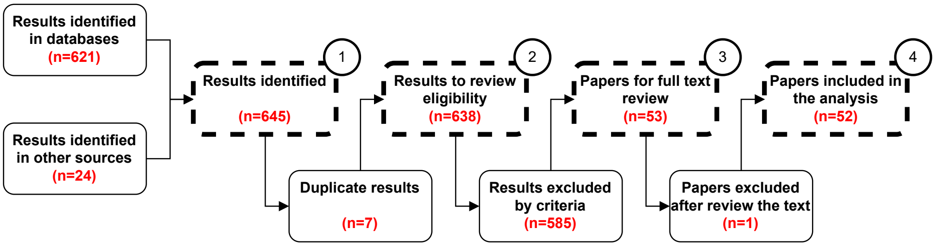


Figure I.2: SLR process summary

considered, or whether the adaptation is done in real-time. In this way, a score is obtained for each item analyzed.

Table I.4: SLR Data extraction quality assessment checklist

Question
Is the proposal (architecture/framework) described?
Are the users' needs considered?
Are the customs situations considered?
Is the proposal implemented?
Is the proposal based on a standard?
Is the interoperability improved?
Is the proposal validated? (examples, use cases, etc)
Is the adaptation to the situation performed in real-time?

(a) Questions

Answer	Weight
Yes	1.0
Partially	0.5
Undefined	0.0
No	-0.2

(b) Answers

On the other hand, a data extraction form was elaborated, which is filled in for each paper (Table I.5). The form is composed of different fields to be filled in with the relevant information for the SLR. Among some of the fields to be filled in are whether the paper is based on a standard, whether different situations in IoT environments are considered, the data source of the devices, or the application domain.

With all this, the results are obtained, which will later be analyzed to determine the path to follow during the thesis.

Table I.5: SLR Data extraction form

Description	Type	Values
Based on a standard	Select One Field	• No • Partially • Yes
If based in a standard, describe which one(s)	String Field	• n/a
Considers environment's situations	Boolean Field	• n/a
Adaptation in real-time Select	One Field	• No • Partially • Yes
Solution definition (architecture/framework)	Boolean Field	• n/a
IoT devices data source	Select One Field	• Not specified • Real • Synthetic
IoT/WoT domain	Select Many Field	• Agriculture • Automotive • Healthcare • Industry • Mobility • Smart city • Smart grid • Smart home/office • Sports • Another
If other domain, describe which one(s)	String Field	• n/a
Interoperability reached	Select Many Field	• IoT/WoT Domains • Organizational • Other • Semantics • Situational • Syntax • Technology
If other interoperability level, describe which one(s)	String Field	• n/a
Pros+	String Field	• n/a
Cons-	String Field	• n/a
Notes	String Field	• n/a

Acronyms

AmI Ambient Intelligent. 6, 35–37

API Application Programming Interface. 68, 78, 80, 86, 92

BLE Bluetooth Low Energy. 72

COP Context-Oriented Programming. 6, 23–26

GGS GII-GRIN-SCIE. 14

HASS Home Assistant. 61, 68, 72

HDM Human Data Model. 78–81, 90

IoB Internet of Bodies. 78, 80, 81

IoP Internet of People. 2

IoT Internet of Things. xv, 4, 5, 7–13, 16, 17, 20, 22, 23, 26, 28, 34, 37, 38, 42, 45, 46, 48–51, 55, 58, 59, 64, 66, 69, 78, 80, 81, 83, 86, 90–93, 98, 102, 112, 115–118

JCR Journal Citation Reports. 14, 15

ML Machine Learning. 6

MQTT Message Queuing Telemetry Transport. 55, 72, 73, 96

NMAP Network Mapper. 72

OWL Web Ontology Language. 26, 29

PeaaS People as a Service. xv, 2, 13, 32–35

RDF Resource Description Framework. 26, 27

REST REpresentational State Transfer. 34

RFID Radio Frequency Identification. 7

SLR Systematic Literature Review. xv, 38, 39, 41, 42, 44, 51, 55

SOAP Simple Object Access Protocol. 34

SPARQL SPARQL Protocol and RDF Query Language. 69, 74

Spilab Social and Pervasive Innovation Lab. 2, 3

SW Semantic Web. 6

URI Uniform Resource Identifiers. 26

W3C World Wide Web Consortium. 26

WoT Web of Things. 65, 115, 116

References

Abbasi, M. A., Memon, Z. A., Durrani, N. M., Haider, W., Laeeq, K., & Mallah, G. A. (2021). A multi-layer trust-based middleware framework for handling interoperability issues in heterogeneous iots. *Cluster Computing*, *24*(3), 2133–2160.

Adam, M. S., Anisi, M. H., & Ali, I. (2020). Object tracking sensor networks in smart cities: Taxonomy, architecture, applications, research challenges and future directions. *Future Generation Computer Systems*, *107*, 909–923.

Alessi, M., Giangreco, E., Pinnella, M., Pino, S., Storelli, D., Mainetti, L., ... Patrono, L. (2016, aug). A web based virtual environment as a connection platform between people and IoT. In *2016 international multidisciplinary conference on computer and energy science, splitech 2016.* Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/SpliTech.2016.7555925

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015a). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, *17*(4), 2347–2376.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015b, oct). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys and Tutorials*, *17*(4), 2347–2376. doi: 10.1109/COMST.2015.2444095

Ali, S., Kibria, M. G., & Chong, I. (2017, apr). WoO enabled IoT service provisioning based on learning user preferences and situation. *International Conference on Information Networking*, 474–476. doi: 10.1109/ICOIN.2017.7899538

Amarilli, F., Amigoni, F., Fugini, M. G., & Zarri, G. P. (2017, feb). A Semantic-Rich Approach to IoT Using the Generalized World Entities Paradigm. In *Managing the web of things: Linking the real world to the web* (pp. 105–147). Elsevier Inc. doi: 10.1016/B978-0-12-809764-9.00005-6

André, P., Azzi, F., & Cardin, O. (2019). Heterogeneous communication middleware for digital twin based cyber manufacturing systems. In *International workshop on service orientation in holonic and multi-agent manufacturing* (pp. 146–157).

Appeltauer, M., Hirschfeld, R., Haupt, M., Lincke, J., & Perscheid, M. (2009). A comparison of context-oriented programming languages. In *International workshop on context-oriented programming* (pp. 1–6).

Arcaini, P., Mirandola, R., Riccobene, E., Scandurra, P., Arrigoni, A., Bosc, D., ... Pedercini, R. (2020, mar). Smart home platform supporting decentralized adaptive automation control. *Proceedings of the ACM Symposium on Applied Computing*, 1893–1900. doi: 10.1145/3341105.3373925

Asghar, M. H., Negi, A., & Mohammadzadeh, N. (2015, jul). Principle application and vision in Internet of Things (IoT). In *International conference on computing, communication and automation, iccca 2015* (pp. 427–431). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/CCAA.2015.7148413

Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer networks*, *54*(15), 2787–2805.

Barnaghi, P., Wang, W., Henson, C., & Taylor, K. (2012a, jan). Semantics for the Internet of Things. *International Journal on Semantic Web & Information Systems*, *8*(1), 1–21. Retrieved from https://dl.acm.org/doi/abs/10.4018/jswis.2012010101 doi: 10.4018/JSWIS.2012010101

Barnaghi, P., Wang, W., Henson, C., & Taylor, K. (2012b). Semantics for the internet of things: early progress and back to the future. *International Journal on Semantic Web and Information Systems (IJSWIS)*, *8*(1), 1–21.

Berners-Lee, T. (2000). *Semantic web on xml.* https://www.w3.org/2000/Talks/1206-xml2k-tbl/Overview.html. ((Accessed on

15/02/2022))

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific american*, *284*(5), 34–43.

Berrocal, J., Garcia-Alonso, J., Canal, C., & Murillo, J. M. (2016). Situational-context: a unified view of everything involved at a particular situation. In *International conference on web engineering* (pp. 476–483).

Berrocal, J., Garcia-Alonso, J., Murillo, J. M., & Canal, C. (2017). Rich contextual information for monitoring the elderly in an early stage of cognitive impairment. *Pervasive and Mobile Computing*, *34*, 106–125.

Bonino, D., Alizo, M. T. D., Alapetite, A., Gilbert, T., Axling, M., Udsen, H., ... Spirito, M. (2015, oct). ALMANAC: Internet of things for smart cities. In *Proceedings - 2015 international conference on future internet of things and cloud, ficloud 2015 and 2015 international conference on open and big data, obd 2015* (pp. 309–316). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/FiCloud.2015.32

Busold, C., Heuser, S., Rios, J., Sadeghi, A. R., & Asokan, N. (2015). Smart and secure cross-device Apps for the Internet of advanced things. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8975*, 272–290. doi: 10.1007/978-3-662-47854-7_17

Catania, V., La Delfa, G. C., Monteleone, S., Patti, D., Ventura, D., & La Torre, G. (2019). GOOSE: Goal oriented orchestration for smart environments. *International Journal of Ad Hoc and Ubiquitous Computing*, *32*(3), 159–170. doi: 10.1504/IJAHUC.2019.103130

Chen, S., Liu, T., Gao, F., Ji, J., Xu, Z., Qian, B., ... Guan, X. (2017). Butler, not servant: A human-centric smart home energy management system. *IEEE Communications Magazine*, *55*(2), 27–33.

Cheng, B., Wang, M., Zhao, S., Zhai, Z., Zhu, D., & Chen, J. (2017, aug). Situation-Aware Dynamic Service Coordination in an IoT Environment. *IEEE/ACM Transactions on Networking*, *25*(4), 2082–2095. doi: 10.1109/TNET.2017.2705239

Cheng, B., Zhu, D., Zhao, S., & Chen, J. (2016, jun). Situation-aware IoT service coordination using the event-driven SOA paradigm. *IEEE*

Transactions on Network and Service Management, *13*(2), 349–361. doi: 10.1109/TNSM.2016.2541171

Chiang, M., & Zhang, T. (2016). Fog and iot: An overview of research opportunities. *IEEE Internet of Things Journal*, *3*(6), 854–864.

Čolaković, A., & Hadžialić, M. (2018, oct). *Internet of Things (IoT): A review of enabling technologies, challenges, and open research issues* (Vol. 144). Elsevier B.V. doi: 10.1016/j.comnet.2018.07.017

Conti, M., Passarella, A., & Das, S. K. (2017). The internet of people (IoP): A new wave in pervasive mobile computing. *Pervasive and Mobile Computing*, *41*, 1–27.

Cook, D. J., Augusto, J. C., & Jakkula, V. R. (2009). Ambient intelligence: Technologies, applications, and opportunities. *Pervasive and Mobile Computing*, *5*(4), 277–298.

Cubo, J., Brogi, A., & Pimentel, E. (2012). Towards behaviour-aware compositions of things in the future internet. *ACM International Conference Proceeding Series*, 28–35. doi: 10.1145/2377836.2377844

Davoudpour, M., Sadeghian, A., & Rahnama, H. (2015, nov). Synthesizing social context for making Internet of Things environments more immersive. *2015 International Conference on the Network of the Future, NOF 2015*. doi: 10.1109/NOF.2015.7333282

De Matos, E., Amaral, L. A., Tiburski, R. T., Schenfeld, M. C., De Azevedo, D. F., & Hessel, F. (2017, jul). A sensing-as-a-service context-aware system for Internet of Things environments. In *2017 14th ieee annual consumer communications and networking conference, ccnc 2017* (pp. 724–727). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/CCNC.2017.7983223

Elkhodr, M., Shahrestani, S., & Cheung, H. (2016). The internet of things: new interoperability, management and security challenges. *arXiv preprint arXiv:1604.04824*.

Feng, S., Setoodeh, P., & Haykin, S. (2017). Smart home: Cognitive interactive people-centric internet of things. *IEEE Communications Magazine*, *55*(2), 34–39.

Flores-Martin, D., Berrocal, J., García-Alonso, J., & Murillo, J. M. (2020). Extending w3c thing description to provide support for interactions of things in real-time. In *International conference on web engineering* (pp. 30–41).

Flores-Martin, D., Berrocal, J., García-Alonso, J., & Murillo, J. M. (2022). Towards dynamic and heterogeneous social iot environments. *Computing*, 1–24.

Flores-Martin, D., Pérez-Vereda, A., Berrocal, J., Canal, C., & Murillo, J. M. (2019, dec). Interconnecting IoT devices to improve the qol of elderly people. In *Communications in computer and information science* (Vol. 1016, pp. 83–93). Springer Verlag. Retrieved from https://doi.org/10.1007/978-3-030-16028-9_8 doi: 10.1007/978-3-030-16028-9_8

Giordano, A., & Spezzano, G. (2014). Service-oriented middleware for the cooperation of smart objects and web services. *Internet of Things*, *0*(9783319004907), 49–68. doi: 10.1007/978-3-319-00491-4_3

Gomez, C., Chessa, S., Fleury, A., Roussos, G., & Preuveneers, D. (2019, jan). Internet of Things for enabling smart environments: A technology-centric perspective. *Journal of Ambient Intelligence and Smart Environments*, *11*(1), 23–43. doi: 10.3233/AIS-180509

Guillen, J., Miranda, J., Berrocal, J., Garcia-Alonso, J., Murillo, J. M., & Canal, C. (2013). People as a service: a mobile-centric model for providing collective sociological profiles. *IEEE software*, *31*(2), 48–53.

Gusmeroli, S., Haller, S., Harrison, M., Kalaboukas, K., Tomasella, M., Vermesan, O., ... Wouters, K. (2010). Vision and Challenges for Realising the Internet of Things. *undefined*.

Gyrard, A., Datta, S. K., Bonnet, C., & Boudaoud, K. (2015a, Aug). Cross-domain internet of things application development: M3 framework and evaluation. In *2015 3rd international conference on future internet of things and cloud* (p. 9-16).

Gyrard, A., Datta, S. K., Bonnet, C., & Boudaoud, K. (2015b). A semantic engine for internet of things: cloud, mobile devices and gateways. In *Innovative mobile and internet services in ubiquitous computing (imis), 2015 9th international conference on* (pp. 336–341).

Hirschfeld, R., Costanza, P., & Nierstrasz, O. M. (2008). Context-oriented programming. *Journal of Object technology*, *7*(3), 125–151.

Howarth, J. (2022). *80+ amazing iot statistics (2022-2030).* https://explodingtopics.com/blog/iot-stats. ((Accessed on 05/09/2022))

Http vs. mqtt: A tale of two iot protocols. (2018). `https://cloud.google.com/blog/products/iot-devices/http-vs-mqtt-a-tale-of-two-iot-protocols`.

Hussain, A., Wenbi, R., Da Silva, A. L., Nadher, M., & Mudhish, M. (2015, dec). Health and emergency-care platform for the elderly and disabled people in the Smart City. *Journal of Systems and Software*, *110*, 253–263. doi: 10.1016/j.jss.2015.08.041

IEEE. (2022). *Ieee internet of things.* `https://iot.ieee.org/`. ((Accessed on 17/01/2022))

Insights, T. (2020). *Global iot market to grow to 24.1 billion devices in 2030, generating $1.5 trillion annual revenue.* `https://transformainsights.com/news/iot-market-24-billion-usd15-trillion-revenue-2030`. ((Accessed on 16/01/2022))

Javed, B., Iqbal, M. W., & Abbas, H. (2017, jun). Internet of things (IoT) design considerations for developers and manufacturers. In *2017 ieee international conference on communications workshops, icc workshops 2017* (pp. 834–839). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICCW.2017.7962762

Ji, B., Wang, Y., Song, K., Li, C., Wen, H., Menon, V. G., & Mumtaz, S. (2021). A survey of computational intelligence for 6g: Key technologies, applications and trends. *IEEE Transactions on Industrial Informatics*, *17*(10), 7145–7154.

Kaebisch, S., Kamiya, T., McCool, M., & Charpenay, V. (2019). *Web of things (wot) thing description. candidate recommendation, w3c.*

Keays, R., & Rakotonirainy, A. (2003). Context-oriented programming. In *Proceedings of the 3rd acm international workshop on data engineering for wireless and mobile access* (pp. 9–16).

Khamparia, A., & Pandey, B. (2017). Comprehensive analysis of semantic web reasoners and tools: a survey. *Education and Information Technologies*, *22*(6), 3121–3145.

Khan, R., Khan, S. U., Zaheer, R., & Khan, S. (2012). Future internet: The internet of things architecture, possible applications and key challenges. In *Proceedings - 10th international conference on frontiers of information technology, fit 2012* (pp. 257–260). doi: 10.1109/FIT.2012.53

Kim, J., Yun, J., Choi, S.-C., Seed, D. N., Lu, G., Bauer, M., ... Song, J. (2016). Standard-based iot platforms interworking: implementation, experiences, and lessons learned. *IEEE Communications Magazine*, *54*(7), 48–54.

Kim, N., Lee, S., & Ha, T. (2015, aug). Understanding IoT Through the Human Activity: Analogical Interpretation of IoT by Activity Theory. *Communications in Computer and Information Science*, *528*, 38–42. Retrieved from https://link.springer.com/chapter/10.1007/978-3-319-21380-4_7 doi: 10.1007/978-3-319-21380-4_7

Kitchenham, B., & Charters, S. (2007). *Guidelines for performing systematic literature reviews in software engineering.*

Klusch, M., Kapahnke, P., Schulte, S., Lecue, F., & Bernstein, A. (2016). Semantic web service search: a brief survey. *KI-Künstliche Intelligenz*, *30*(2), 139–147.

Laso, S., Berrocal, J., García-Alonso, J., Canal, C., & Manuel Murillo, J. (2021). Human microservices: A framework for turning humans into service providers. *Software: Practice and Experience*, *51*(9), 1910–1935.

Li, S., Xu, L. D., & Zhao, S. (2015). The internet of things: a survey. *Information systems frontiers*, *17*(2), 243–259.

Maarala, A. I., Su, X., & Riekki, J. (2017). Semantic reasoning for context-aware internet of things applications. *IEEE Internet of Things Journal*, *4*(2), 461–473.

Maedche, A., & Staab, S. (2001). Ontology learning for the semantic web. *IEEE Intelligent systems*, *16*(2), 72–79.

Maheswaran, M., & Misra, S. (2015). Towards a social governance framework for Internet of Things. *IEEE World Forum on Internet of Things, WF-IoT 2015 - Proceedings*, 801–806. doi: 10.1109/WF-IOT.2015.7389156

Maita-Tepán, X., Garnica-Bautista, X., & Cedillo, P. (2019). Towards an architecture for human computer interaction within the internet of things and ambient assisted living domain. *XXII Ibero-American Conference on Software Engineering, CIbSE 2019*, 265–278.

Mäkitalo, N., Flores-Martin, D., Flores, H., Lagerspetz, E., Christophe, F., Ihantola, P., ... others (2020). Human data model: Improving programmability of health and well-being data for enhanced perception and interaction. *ACM Transactions on Computing for Healthcare*, *1*(4), 1–39.

Matwyshyn, A. M. (2019). The internet of bodies. *Wm. & Mary L. Rev.*, *61*, 77.

Merkle, N., & Zander, S. (2016, jan). Improving the Utilization of AAL Devices through Semantic Web Technologies and Web of Things Concepts. In *Procedia computer science* (Vol. 58, pp. 290–297). Elsevier B.V. doi: 10.1016/j.procs.2016.09.045

Miori, V., Russo, D., & Ferrucci, L. (2019). Interoperability of home automation systems as a critical challenge for iot. In *2019 4th international conference on computing, communications and security, icccs.*

Miranda, J., Mäkitalo, N., Garcia-Alonso, J., Berrocal, J., Mikkonen, T., Canal, C., & Murillo, J. M. (2015). From the internet of things to the internet of people. *IEEE Internet Computing*, *19*(2), 40–47.

Mishra, R. B., & Kumar, S. (2011). Semantic web reasoners and languages. *Artificial Intelligence Review*, *35*(4), 339–368.

Mongiello, M., Boggia, G., & Di Sciascio, E. (2016). Reios: Reflective architecting in the internet of objects. In *2016 4th international conference on model-driven engineering and software development (modelsward)* (pp. 384–389).

Morabito, R., Petrolo, R., Loscri, V., & Mitton, N. (2018). Legiot: A lightweight edge gateway for the internet of things. *Future Generation Computer Systems*, *81*, 1–15.

Motta, R. C., De Oliveira, K. M., & Travassos, G. H. (2017, jun). Rethinking Interoperability in Contemporary Software Systems. *Proceedings - 2017 IEEE/ACM Joint 5th International Workshop on Software Engineering for Systems-of-Systems and 11th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems, JSOS 2017*, 9–15. doi: 10.1109/JSOS.2017.5

Mäkitalo, N. (2019). *What is human data model?* `https://humandatamodel.github.io/`.

Noura, M., Atiquzzaman, M., & Gaedke, M. (2019a). Interoperability in internet of things: Taxonomies and open challenges. *Mobile Networks and Applications*, *24*(3), 796–809.

Noura, M., Atiquzzaman, M., & Gaedke, M. (2019b, jun). Interoperability in Internet of Things: Taxonomies and Open Challenges. *Mobile Networks and Applications*, *24*(3), 796–809. Retrieved from https://doi.org/10.1007/s11036-018-1089-9 doi: 10.1007/s11036-018-1089-9

Noura, M., Heil, S., & Gaedke, M. (2018, jun). GrOWTH: Goal-oriented end user development for web of things devices. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)* (Vol. 10845 LNCS, pp. 358–365). Springer Verlag. Retrieved from https://doi.org/10.1007/978-3-319-91662-0_29 doi: 10.1007/978-3-319-91662-0_29

Nunes, L. H., Estrella, J. C., Perera, C., Reiff-Marganiec, S., & Delbem, A. C. (2018). The elimination-selection based algorithm for efficient resource discovery in internet of things environments. In *Consumer communications & networking conference (ccnc), 2018 15th ieee annual* (pp. 1–7).

Ortiz, G., Zouai, M., Kazar, O., Garcia-de Prado, A., & Boubeta-Puig, J. (2022). Atmosphere: Context and situational-aware collaborative iot architecture for edge-fog-cloud computing. *Computer Standards & Interfaces*, *79*, 103550.

Pantsar-Syväniemi, S., Purhonen, A., Ovaska, E., Kuusijärvi, J., & Evesti, A. (2012, jan). Situation-based and self-adaptive applications for the smart environment. *Journal of Ambient Intelligence and Smart Environments*, *4*(6), 491–516. doi: 10.3233/AIS-2012-0179

Patel, K. K., Patel, S. M., et al. (2016). Internet of things-iot: definition, characteristics, architecture, enabling technologies, application & future challenges. *International journal of engineering science and computing*, *6*(5).

Perera, C., Zaslavsky, A., Christen, P., & Georgakopoulos, D. (2014). Context aware computing for the internet of things: A survey. *IEEE Communications Surveys & Tutorials*, *16*(1), 414–454.

Pérez-Vereda, A., & Canal, C. (2017). A people-oriented paradigm for smart cities. In *International conference on web engineering* (pp. 584–591).

Perwej, Y., Haq, K., Parwej, F., & M., M. (2019, apr). The Internet of Things (IoT) and its Application Domains. *International Journal of Computer Applications*, *182*(49), 36–49. doi: 10.5120/IJCA2019918763

Pötter, H. B., & Sztajnberg, A. (2016, may). Adapting heterogeneous devices into an IoT context-aware infrastructure. In *Proceedings - 11th international symposium on software engineering for adaptive and self-managing systems, seams 2016* (pp. 64–74). Association for Computing Machinery, Inc. doi: 10.1145/2897053.2897072

Pradeep, P., Krishnamoorthy, S., & Vasilakos, A. V. (2021). A holistic approach to a context-aware iot ecosystem with adaptive ubiquitous middleware. *Pervasive and Mobile Computing*, *72*, 101342.

Rakib, A., & Uddin, I. (2019, feb). An Efficient Rule-Based Distributed Reasoning Framework for Resource-bounded Systems. *Mobile Networks and Applications*, *24*(1), 82–99. doi: 10.1007/s11036-018-1140-x

RedAlkemi. (2018). *Pros & cons of internet of things.* `https://www.redalkemi.com/blog/post/internet-of-things-applications`. ((Accessed on 16/01/2022))

Rentero-Trejo, R., Flores-Martín, D., García-Alonso, J., Galán-Jiménez, J., & Murillo, J. (2021). Using federated learning to achieve proactive context-aware iot environments. *Journal of Web Engineering*, *21*(1), 53–74.

Rhayem, A., Mhiri, M. B. A., & Gargouri, F. (2020). Semantic web technologies for the internet of things: Systematic literature review. *Internet of Things*, *11*, 100206.

Rojo, J., Flores-Martin, D., Garcia-Alonso, J., Murillo, J. M., & Berrocal, J. (2020). Automating the interactions among iot devices using neural networks. In *2020 ieee international conference on pervasive computing and communications workshops (percom workshops)* (pp. 1–6).

Roman, R., Zhou, J., & Lopez, J. (2013). On the features and challenges of security and privacy in distributed internet of things. *Computer Networks*, *57*(10), 2266–2279.

Ruta, M., Scioscia, F., Loseto, G., & Di Sciascio, E. (2017). A semantic-enabled social network of devices for building automation. *IEEE Transactions on Industrial Informatics*, *13*(6), 3379–3388.

Said, O., & Masud, M. (2013). Towards Internet of Things: Survey and Future Vision. *Omar Said & Mehedi Masud International Journal of Computer Networks (IJCN)*(5), 1.

Salvaneschi, G., Ghezzi, C., & Pradella, M. (2012). Context-oriented programming: A software engineering perspective. *Journal of Systems and Software*, *85*(8), 1801–1817.

Saputra, Y., Hua, J., Wendt, N., Julien, C., & Roman, G.-C. (2019). Warble: Programming abstractions for personalizing interactions in the internet of things. In *2019 ieee/acm 6th international conference on mobile software engineering and systems (mobilesoft)* (pp. 128–139).

Sasirekha, S., & Swamynathan, S. (2016, sep). Collaboration of IoT devices using semantically enabled resource oriented middleware. In *Acm international conference proceeding series* (Vol. 21-24-Sept, pp. 98–105). Association for Computing Machinery. Retrieved from `http://dl.acm.org/citation.cfm?doid=2983402.2983428` doi: 10.1145/2983402.2983428

Saxena, P. (2016). *The advantages and disadvantages of internet of things.* `https://e27.co/advantages-disadvantages-internet-things-20160615/`. ((Accessed on 16/01/2022))

Schmidt, D. C., Stal, M., Rohnert, H., & Buschmann, F. (2013). *Pattern-oriented software architecture, patterns for concurrent and networked objects.* John Wiley & Sons.

Scioscia, F., Ruta, M., Loseto, G., Gramegna, F., Ieva, S., Pinto, A., & Di Sciascio, E. (2014). A mobile matchmaker for the ubiquitous semantic web. *International Journal on Semantic Web and Information Systems (IJSWIS)*, *10*(4), 77–100.

Seeliger, A., Pfaff, M., & Krcmar, H. (2019). Semantic web technologies for explainable machine learning models: A literature review. *PROFILES/SEMEX@ ISWC*, *2465*, 1–16.

Seydoux, N., Drira, K., Hernandez, N., & Monteil, T. (2016). Iot-o, a core-domain iot ontology to represent connected devices networks. In *European knowledge acquisition workshop* (pp. 561–576).

Sezer, O. B., Dogdu, E., & Ozbayoglu, A. M. (2018, feb). *Context-Aware Computing, Learning, and Big Data in Internet of Things: A Survey* (Vol. 5) (No. 1). Institute of Electrical and Electronics Engineers Inc.

doi: 10.1109/JIOT.2017.2773600

Shrestha, N., Kubler, S., & Främling, K. (2014). Standardized framework for integrating domain-specific applications into the iot. In *Future internet of things and cloud (ficloud), 2014 international conference on* (pp. 124–131).

Taivalsaari, A., & Mikkonen, T. (2017). A roadmap to the programmable world: software challenges in the iot era. *IEEE Software*, *34*(1), 72–80.

Tayur, V. M., & Suchithra, R. (2017, jul). Review of interoperability approaches in application layer of Internet of Things. *IEEE International Conference on Innovative Mechanisms for Industry Applications, ICIMIA 2017 - Proceedings*, 322–326. doi: 10.1109/ICIMIA.2017.7975628

Techopedia. (2017). *Semantic web.* `https://www.techopedia.com/definition/27961/semantic-web`. ((Accessed on 14/02/2022))

Temglit, N., Chibani, A., Djouani, K., & Nacer, M. A. (2016, jan). Distributed Approach for QoS Service Selection in Web of Objects. In *Procedia computer science* (Vol. 83, pp. 1170–1175). Elsevier. doi: 10.1016/j.procs.2016.04.240

Thomas, I., Fedon, L., Jara, A., & Bocchi, Y. (2015, sep). Towards a Human Centric Intelligent Society: Using Cloud and the Web of Everything to Facilitate New Social Infrastructures. *Proceedings - 2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2015*, 319–324. doi: 10.1109/IMIS.2015.94

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *The knowledge engineering review*, *11*(2), 93–136.

van der Schaaf, H., Moßgraber, J., Grellet, S., Beaufils, M., Schleidt, K., & Usländer, T. (2020, feb). An Environmental Sensor Data Suite Using the OGC SensorThings API. In *Ifip advances in information and communication technology* (Vol. 554 IFIP, pp. 228–241). Springer. doi: 10.1007/978-3-030-39815-6_22

Van Kranenburg, R., & Bassi, A. (2012). Iot challenges. *Communications in Mobile Computing*, *1*(1), 9.

Venceslau, A. D., Andrade, R. M., Vidal, V. M., Nogueira, T. P., & Pequeno, V. M. (2019). IoT semantic interoperability: A systematic mapping study. In *Iceis 2019 - proceedings of the 21st international conference on enterprise information systems* (Vol. 1, pp. 523–532). SciTePress. doi: 10.5220/0007732605350544

Vinob chander, R. (2010). *Novel ubiquitous interoperable context-aware smart environments through web services.*

Yacchirema, D. C., Palau, C. E., & Esteve, M. (2017). Enable iot interoperability in ambient assisted living: Active and healthy aging scenarios. In *2017 14th ieee annual consumer communications & networking conference (ccnc)* (pp. 53–58).

Yafei, D., Guanyu, L., & Hui, Z. (2016). Semantic space-based semantic collaboration method in semantic web of things. *Computer Applications and Software*, *2*, 002.

Yaqoob, I., Ahmed, E., Hashem, I. A. T., Ahmed, A. I. A., Gani, A., Imran, M., & Guizani, M. (2017). Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges. *IEEE Wireless Communications*, *24*(3), 10–16. doi: 10.1109/MWC.2017.1600421

Yu, T., Li, T., Sun, Y., Nanda, S., Smith, V., Sekar, V., & Seshan, S. (2020). Learning context-aware policies from multiple smart homes via federated multi-task learning. In *2020 ieee/acm fifth international conference on internet-of-things design and implementation (iotdi)* (pp. 104–115).

Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., & Gao, Y. (2021). A survey on federated learning. *Knowledge-Based Systems*, *216*, 106775.

Zhou, J., Cao, Z., Dong, X., & Vasilakos, A. V. (2017). Security and privacy for cloud-based iot: challenges. *IEEE Communications Magazine*, *55*(1), 26–33.