UNIVERSIDAD DE EXTREMADURA

Escuela Politécnica

Máster en Ingeniería Informática

Trabajo de Fin de Máster

Towards a Runtime Interconnection in a Multi-device IoT Environment

Daniel Flores Martín

Febrero, 2019

UNIVERSIDAD DE EXTREMADURA

Escuela Politécnica

Máster en Ingeniería Informática

Trabajo de Fin de Máster

Towards a Runtime Interconnection in a Multi-device IoT Environment

Autor: Daniel Flores Martín
Fdo:

Director: Juan Manuel Murillo Rodríguez
Fdo:

Co-director: José Javier Berrocal Olmeda
Fdo:

Tribunal Calificador

Presidente: Pedro José Clemente Martín
Fdo:

Secretario: José Manuel García Alonso
Fdo:

Vocal: Marino Linaje Trigueros
Fdo:

Acknowledgements

First of all, I would like to thank my directors Juanma and Javi, for the effort and dedication they have shown me during the realization of this work. Their knowledge, work habits, discipline, persistence and infinite patience have been fundamental for me throughout the development of this work.

Second, to Quercus Software Engineering Group of the University of Extremadura, to which it is an honour to belong for all the good things it is providing me in my scientific career, as well as for the people who compose it, who are excellent professionals and very nice people.

And last but not least, to my family and friends. Thank you for your unconditional support and the motivation that has been so necessary during all this time, both for the good and for the less good.

Thanks

Resumen

Hoy en día, hay millones de dispositivos conectados a Internet. Esto es lo que conocemos como la Internet de las Cosas. El propósito de estos dispositivos es hacer la vida de las personas más fácil. Pero el real potencial de este paradigma se consigue gracias a la colaboración entre dispositivos. Sin embargo, muchos fabricantes desarrollan sus propios dispositivos y protocolos de forma endogámica para proteger su cuota de mercado, limitando en muchos sentidos la colaboración o coordinación entre dispositivos de diferentes fabricantes. Además, su adaptación al contexto es importante para solventar las necesidades detectadas. Esta adaptación se ha hecho tradicionalmente a través de la configuración manual por parte de los usuarios, con la consiguiente inversión de tiempo y esfuerzo. Este trabajo de fin de máster presenta una solución basada, por un lado, en técnicas de web semántica con el objetivo de permitir la comunicación entre dispositivos independientemente de los lenguajes y protocolos desarrollados por sus fabricantes; y por otro, en técnicas de aprendizaje automático con el objetivo de aprender de las rutinas de las personas para automatizar el comportamiento de los dispositivos. Con esta propuesta, dispositivos de diferentes fabricantes pueden comunicarse para crear un entorno colaborativo de forma sencilla, eficiente y perfectamente asequible, además de adquirir inteligencia adicional que les permita adaptar su comportamiento a las preferencias de las personas de forma automática.

Palabras clave— Internet de las Cosas, Colaboración de dispositivos, Adaptación al contexto, Necesidades de las personas

Abstract

Nowadays, there are millions of devices connected to the Internet. This is what we know as the Internet of Things. The purpose of these devices is to make people's lives easier. Thanks to the collaboration between devices, the possibilities offered by the Internet of Things can be exploited even more. However, many manufacturers develop their own devices and protocols endogamically to protect their market share, limiting the collaboration and coordination between devices from different manufacturers. In addition, devices behaviour adaptation to the context is important in order to meet the identified needs. This adaptation has traditionally been done through manual configuration by users, with the consequent investment of time and effort. This master thesis presents a solution based on, on the one hand, semantic web techniques with the aim of achieving collaboration between devices independently of the languages and protocols developed by their manufacturers;and, on the other hand, machine learning techniques with the aim of learning from people's routines to automate the behaviour of the devices. With this proposal, devices from different manufacturers can communicate to create a collaborative environment in a simple, efficient and perfectly affordable way, acquiring additional intelligence that allows them to adapt their behavior according to people's preferences and depending on the context in an automatically way.

Keywords— Internet of Things, Devices collaboration, Context adaptation, People's needs

Contents

List of Figures

Chapter 1

Introduction

Contents

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. Today, we can connect to the Internet our mobile phones, printers, smart TVs, IP cameras, GPS and a multitude of electronic devices that have this functionality, but have we reached the limit or is it just the beginning of a new era of the Internet?

A few years ago 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. 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 automatically.

The great evolution of the IoT allow us to develop increasingly smart devices and truly useful services that a few years ago were unthinkable, such as lighting or temperature controller, devices checking the blood pressure, or detecting and notifying emergency situations. In order to get the maximum benefit from them, they must collaborate with each other, to perform complex tasks [3]. So, the next evolution of the IoT is to ensure that smart devices can proactively collaborate [4, 5], as it is also pursued in the programmable world [6].

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 in terms of collaborating, which means that they can interact but they can not to be integrated with devices from other manufactures. 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 [7]. 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.

This collaboration should not be random or fortuitous, it should come directed by the context of the device and the users. The context in which the devices are located is important and provides valuable information about other devices, people data, meteorological factors, date and time, and so on. For this reason, the behaviour of the devices must take into account the characteristics of the context if we want to achieve the most precise adaptation to the people needs. The development of context-sensitive software has proved successful [8]. 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 [9, 10], who often do not have the necessary technical skills, with the consequent investment of time and frustration that this can cause. To minimize people's interaction with devices, this interconnection must emerge from the context. These drawbacks can be addressed by developing software capable of adapting its behavior to the people needs [8, 6]. In addition, several research areas can contribute to solving them, namely Context Oriented Programming (COP), Ambient Intelligent (AmI), SW and Machine Learning (ML). To learn more about these research areas see the Appendix A.

This master thesis proposes 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. 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 [11]. 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 low cost and effectively. The feasibility of this proposal is evidenced by a proof of concept detailed in the following sections.

This work is based on the paradigm called Situational-Context, which main objective is to achieve the highest level of comfort in a given situation by adapting smart devices behaviour. This concept will be detailed in next sections.

1.1 Motivations

IoT is a novel paradigm that is rapidly gaining ground in people 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. [3]. 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 Fig. 1.1. Recent estimates state that in the next few years we will have about 30 billion smart devices connected to the Internet [12]. 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 [5].

However, the fact remains that there are many challenges and issues related to the use of IoT, and they cannot be ignored [13, 14, 15]. Some of these challenges are:

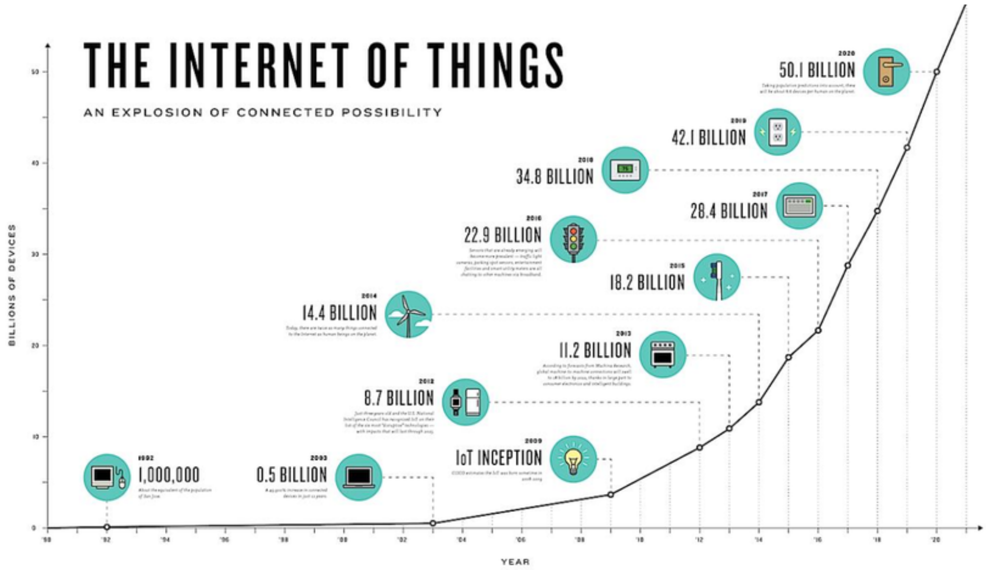


Figure 1.1: Evolution in IoT [1]

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 don't 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 Internet of Things has, as every technology, a number of advantages and disadvantages [16, 17]. Talking about the advantages that the Internet of Things brings to our lives, the most important ones are:

- **Access information:** easy access data and information that is sitting far from the location 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 machine 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 Internet of Things doesn't just bring good things into our lives, there are also some disadvantages that we have 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 Internet of Things that can have a tremendous effect.
- **Lesser jobs:** with every task being automated, the need for human labour 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 lives. There is no doubt that technology is dominating our lifestyle, reflecting a human's dependability on technology.

On the one hand, the Internet of Things allows us to do a lot of interesting things, thanks to the potential of smart devices that we have today, as demonstrated by all its advantages. But on the other hand, there are many disadvantages and problems that presents to be still a new technology and needs to be further researched to take full advantage of its potential. For this reason, the aim of this work is to try to solve the disadvantages shown above, as far as possible, in particular to improving user privacy, promoting compatibility between devices, and reducing people's dependence on smart devices' configuration and adaptation.

This document is structured as follows: Chapter 2 details the different objectives that this work covers; in Chapter 3 we review some of the most relevant literature; then, in Chapter 4 the methodology and the schedule followed are shown. The development carried out are detailed in Chapter 5; in Chapter 6 a sort discussion about the work is done and in Chapter 7 some conclusions and future work are described. Finally, in Chapter 8 a personal reflection about this master thesis is told.

Chapter 2

Objectives

This master thesis aims to achieve that Internet of Things devices adapt their behaviour to people's preferences, by enhancing the interoperability of devices and adapting their behaviour to the characteristics of the context. To do this, they must collaborate with each other achieving the highest level of comfort and according to the contextual information that surrounds them. Therefore, the objectives and sub-objectives of this work are:

1. **Improve technological interoperability.** This implies that devices from different manufacturers as well as from different domains are able to communicate and exchange information in given situations. These results are interesting in environment with multiple smart devices that can collaborate to achieve a common goal.

   (a) **Discover the situation.** A concrete situation involves, on the one hand, devices and people, and on the other hand, the context in which they find themselves. There can be countless situations, so the behaviour of the devices has to take this aspect into account.

2. **Adapt device behavior to people's preferences.** In order to achieve the highest level of comfort for people, intelligent devices must be adapted to their needs.

   (a) **Adapt the devices features to people needs.** Devices already know people needs and, now, can set up their behaviour to best meet them.

(b) **Learn about the situations.** What happens in a situation must be processed with the objective to learn about it and work more precisely in the future.

3. **Ensure users' privacy.** The privacy of users must always be taken into account. If we use user data, we must prevent its safe use and sharing.

(a) **Get people and context data.** The capture of user data passes through the use of the smartphone. Thanks to this device we can capture almost anything that happens around a person and the different situations lived. Each user will store their own information on their device, and will be able to decide what and with whom to share it. As well, context characterises can be detected through the different sensors that devices have.

Chapter 3

State of the Art

Contents

This Chapter reviews the state of the art around the two of the main objectives of this master thesis: interoperability and context awareness.

3.1 Interoperability

Interoperability is one of the big problems at IoT. These problems are partly due to the great diversity of protocols on the market, and the first drawback is detected at the network level, as detailed in the following section. Also, the improvement of interoperability can be achieved through specific techniques such as the Semantic Web, also shown in this section, which allows the relationship between devices belonging to different domains or using different technologies. Communication is fundamental to the Internet of Things. Network technologies allow IoT devices to communicate with other devices, as well as with applications and services running in the cloud. The Internet relies on standardized protocols to ensure that communication between heterogeneous devices can be done safely and reliably.

3.1.1 Semantic Web

The concept Semantic Web was coined by Tim Berners-Lee. 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.

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 (URIs) 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 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 on the information available on the Web.

Therefore, the Semantic Web is a process that allows machines to understand and react quickly to complicated human demands based on their meaning. This type of understanding requires appropriate information sources to be semantically structured, which is a difficult task [18].

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. [19] 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 [20], 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. An interesting semantic engine is designed to be flexible enough to be used in different IoT architectures. Besides, [21] 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. [22] 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 [23].

As we can see, the Semantic Web takes on special relevance within the IoT and its use is widespread in this field.

3.2 Context adaptation

This work is partially framed within context-based software development. In addition to detecting contextual information, it is important that devices use this information to adapt their behaviour and to learn from situations. There are different paradigms that try to develop applications that work on the basis of contextual information and learn about the situations detecting routines or patterns. This makes the behavior of the software in one way or another depending on the characteristics of the environment, which allows a greater ability to cope with different situations to solve.

Some of the most important technologies that are aligned with the objectives of this work are shown below and in Appendix A. These technologies aim to develop context-conscious software, to facilitate its integration with people, or to relate and enrich intelligent devices with additional information.

3.2.1 Context-Awareness

When humans talk with humans, they are able to use implicit situational information, or context, to increase the conversational bandwidth. Unfortunately, this ability to convey ideas does not transfer well to humans interacting with computers. In traditional interactive computing, users have an impoverished mechanism for providing input to computers. By improving the computer's access to context, we increase the richness of communication in human-computer interaction and make it possible to produce more useful computational services. The use of context is increasingly important in the fields of handheld and ubiquitous computing, where the user's context is changing rapidly [24].

In computing, Context-Awareness is referred to the capability of an application or a system to be aware of its physical environment and the situation in order to be able to act and answer in a proactive and intelligent way. Building upon the context acquisition and context management models discussed in literature, numerous context management architectures have been proposed. Although many EU projects and industrial researches have explored context-awareness aspects, the proposed approaches fail to completely offer a generic, scalable and flexible architecture supporting both evolving context models and evolving services and applications. Moreover, an efficient context diffusion and coherent integration into mobile communication services continues to be a challenging research area [25].

The Internet of Things also considers the context as a fundamental part of devices. We can find works such as [26], which analyzes the role of the context and relationships with entities in the IoT in order to discover their characteristics. This work proposes an interesting common ontology context model with the ability to handle uncertainty and temporal aspects of context and Bayesian dynamic networks (DBN) are adopted to reason about these contexts to support the detection of the current situation. In addition, Garrido et al. present a model for the development of context recognition applications using Near Field Communication (NFC). This model is based on the association of services and resources to objects augmented through tags. The different types of services assigned to objects are modeled in a hierarchy that defines the order of execution and customizes user interaction through the use of resources and rules [25]. In addition, addressing sensorization issues within the IoT, [27] proposes a Context-Awareness for Internet of Things (CA4IOT) architecture to help users by automating the task of selecting sensors according to a set of detected needs. Special attention is paid to the automated configuration of filtering, fusion and reasoning mechanisms that can be applied to data streams from sensors collected using selected sensors.

This is only a small sample of the importance of context within the IoT. But the treatment of context is not new. Several technologies have been widely used to allow people to develop context-dependent software, such as Context-

Oriented Programming (Appendix A.1) and Ambient Intelligent (Appendix A.2).

3.3 Situational-Context

The paradigm Situational-Context is a proposal that provide support for analyzing the conditions that exist at a particular time and place in order to predict, at runtime, the expected IoT systems behavior regarding people's needs. This proposal is in current development by the author of this master thesis and his supervisors. Next paragraphs are a sort summary about formal description done by the authors [28].

Situational-Context is composed of entities. These entities can be both IoT devices and people represented through their smartphones, indistinctly. Situational-Context can be defined as the composition of the virtual profiles of all the entities involved in a situation. For a meaningful composition of these profiles, we consider that they contain, at least, the following information:

- A basic profile containing the dated raw information with the entity's status, the relationships with other devices and its history. This profile can be seen as a timeline with the changes and interactions that happened to the entity.
- Social profile. This profile contains the results of high level inferences performed over the Basic Profile.
- The goals detailing the status of the environment desired by the entity. These goals can also be deducted from the Basic and social profiles.
- The skills or capabilities that an entity has to make decisions and perform actions capable of modifying the environment and aimed at achieving goals.

The result of composing the virtual profiles of the involved entities is not only the combined information of all entities. It contains the combined history of the entities ordered in a single timeline, the result of high level inferences performed over the combined virtual profiles, the set of goals of the entities and their Skills. From the combined information of the Situational-Context, strategies to achieve goals based on the present Skills should be identified. These strategies will guide the prediction of the interactions that must emerge from the context. Furthermore, the Situational-Context is a dynamic abstraction of the combined profiles and therefore evolves through time. To analyse the instantaneity of this context, we use the concept of Configuration. A Configuration is the unified and stable view of the virtual profiles of the devices involved in the situation at a specific point in time. When changes in the environment happen, the Configuration is no longer stable and must be updated. Thus, a new configuration must be defined from the updated/new virtual profiles of the devices.

To support the Situational-Context, as described above, there are a number of technological issues that must be resolved. First, which device or devices should compute the Situational-Context? This computation can be done in a Cloud environment. However, in order to reduce the network overhead and thanks to the increased capabilities of smart things, this computation can also be done either by a local device or distributively by a set of local devices. Currently, Multi-Agent Systems can be used to develop self-organised, re-configurable and proactive systems. It should be evaluated what technique is the most appropriate for computing the Situational-Context or even if a combination of different techniques should be applied depending on the size and type of the virtual profiles to combine.

Second, how contextual information can be exchanged? There is a wide range of technologies for this purpose. There are middlewares homogenizing the communications in heterogeneous networks. And there are works, like the SOFIA project, creating a semantic interoperability platform for making the information available for smart services. It is necessary to identify whether they can be used for identifying the devices involved in a configuration and to manage the interactions between them.

Third, how common goals should be agreed for a specific configuration when the devices involved have different or, even, opposed goals? Currently, there are algorithms for negotiating which device should perform a command when in the surrounding there are several that have the capabilities to respond to it. In the Situational-Context, negotiation algorithms are needed to autonomously agree on a common goal.

Fourth, how the strategies to achieve goals should be identified? They can be predefined in order to be triggered depending on the goals to achieve, but again this would compromise the applications flexibility. Should they also emerge from each specific configuration? The Spatial and Temporal reasoning areas have been previously used to get a better understanding of the context in order to make sensible decisions. It should be evaluated whether is possible to infer the strategy to execute from the context using these techniques.

Finally, how the interactions can emerge from the Situational-Context and from each configuration? The Self-Adaptive software systems can modify themselves at run-time and, as detailed above, there are different proposals to develop applications that adapt their behaviour to the context. It should be assessed whether these approaches can be used to develop applications that autonomously compose the devices involved in a Configuration and trigger specific actions depending on the established goals and Strategies.

Due to the detected need about the constantly change in both context and people's preferences, we use the Situational-Context paradigm with which we intend to achieve a better interconnection of IoT devices, and get the maximum benefit by adapting their behavior to people's preferences at runtime. With the exchange of virtual profiles, entities know the goals of devices. This information allows an entity to know if using its abilities can solve the need of another entity. This favors the interaction of entities and allows them to be dependent on the context in which they are, at runtime. This architecture achieves the interconnection of IoT devices at the features level. The interconnection is based on relating the skills of one entity with the goals of another. We know that the goals in an entity arise from the lack of skills when obtaining a desired state in the environment, so we must know how to perform this interconnection and that the goals can be resolved in the best way. Each entity has its own vision of the context, and knows the skills and goals of nearby entities, so that it can interact with them. This is achieved by integrating Situational-Context with Semantic Web and ontologies.

During this section, several research areas contributing to provide a multi-device environment have been reviewed. Most of these paradigms allow us to define behaviors for different scenarios at design time, so the adaptation of the devices is limited to situations that developers have been able to identify, making it impossible to adapt them to other situations that may arise from the context and therefore also prevents this operation from being carried out at runtime. The following section will show the methodology to follow to achieve an adaptation of the devices to the context in real time.

Chapter 4

Methodology

This master thesis pursues three general objectives composed by several sub-objectives. The correct coordination of all of them and how they are approached is as important as the complete planning of the different activities that must be carried out for the correct development of the work. The methodology followed to break down the work into different activities and their sequentiality is detailed below.

For the planning and development of this project it has been decided to adopt the Design Science methodology [29]. Design Science is a methodology that offers specific guidelines for research projects. It is especially used in the disciplines of Engineering and Computer Science, although it is applicable to other areas [30].

The main objective of this methodology is to obtain knowledge of a certain problem and its domain and the contribution of innovative products that minimize the problem [30]. This increase in knowledge and creation of innovative products, that is achieved through the creation of artifacts. An artifact is an object made by people with the intention of being used to solve a practical problem. Artifacts can be sketches, pieces of software, hardware, etc. [31].

Design Science is based on approaching a problem, together with its causes and consequences, as detailed for this work in Chapter 2, and tries to solve them. Fig. 4.1 shows the different activities to be carried out in this research methodology.

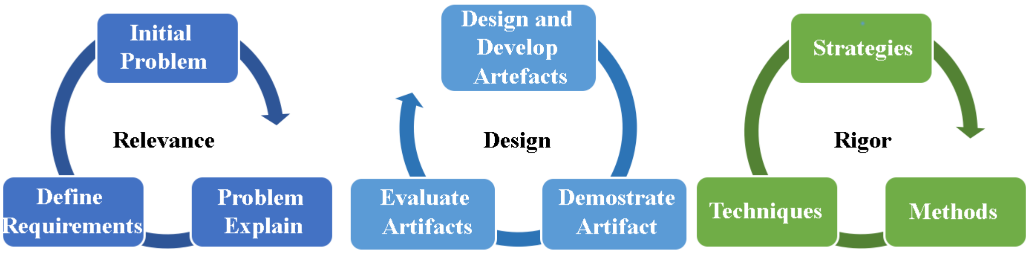


Figure 4.1: Design Science methodology proposed model

- *Problem Explain:* The objective of this activity is to investigate and analyze the initial problem identified as input. The problem must be formulated in a precise and justified manner, highlighting the causes leading to the problem that must be analyzed.
- *Define Requirements:* Sketch the solution to the problem along with the requirements of the solution that will be addressed by the artifacts.
- *Design and Develop Artifacts:* Creation of the artifact that addresses the problem and fulfills the defined objectives.
- *Demonstrate Artifacts:* Use of the artifacts developed in proofs of concept, case studies, etc. that prove the viability of the same one.
- *Evaluate Artifacts:* Determine the extent to which the artifact meets the requirements and solves the problem.

The above activities are not followed strictly sequentially. Normally, and more in research, these tasks are iterative. The arrows between activities only show the expected inputs and outputs of each activity. Therefore, some authors establish a three-cycle view of this methodology [32]. This view is shown in Fig. 4.2.

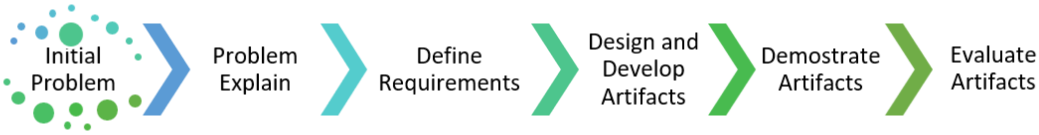


Figure 4.2: Iterative structure of the Design Science methodology

- *Relevance Cycle*, initiates research by analyzing the environment and the problems to be addressed, defining the requirements of the artifacts and the acceptance criteria for the evaluation of the research results. It also identifies whether new iterations are necessary to solve the problem.

- *The Design Cycle*, focuses on the construction and evaluation of the developed artifact. For this purpose, the requirements and criteria identified in the Relevance cycle are followed, and the methods and techniques identified in the Rigor cycle are applied.
- *The Rigor Cycle*, provides knowledge to the project to ensure that it is innovative, ensuring that the research produces contributions to the general knowledge base.

This methodology not only produces artifacts, but also creates knowledge of general interest so it presents three important requirements: first, use of rigorous research methods; second, the results must be correctly grounded and original; third, the results must be communicated.

During the present master thesis, although the research strategy followed to plan this work is Design Science, the research methods that will be used to obtain and analyze the data will be based on different use-cases. The case studies will be used to obtain and validate the artifacts in a broader way.

4.1 Schedule

This master thesis is part of a doctoral thesis. The stages detailed here are a summary of the tasks carried out during that thesis, and which are related to this work at the end of the master's degree. The Fig. 4.3 shows the master thesis task schedule as well as a brief description of each one.

1. **Initial literature review.** During this period, contact is made with the most important technologies that will serve as the basis for the development of the project. This includes the review of works on Context-Awareness, Context-Oriented Programming, Aspect-Oriented Programming and Situational-Context. This revision allows me to know first hand the most important aspects of context-oriented software development, as well as to get in touch with the initial proposal, which will establish one of the most important principles of this work: Situational-Context. Thanks to the work done by Barriga et al. [33], this paradigm already has a sufficiently important base to continue being exploited.
2. **Situational-Context: the architecture.** The development of the architecture begins. In order to do this, a requirements analysis is carried

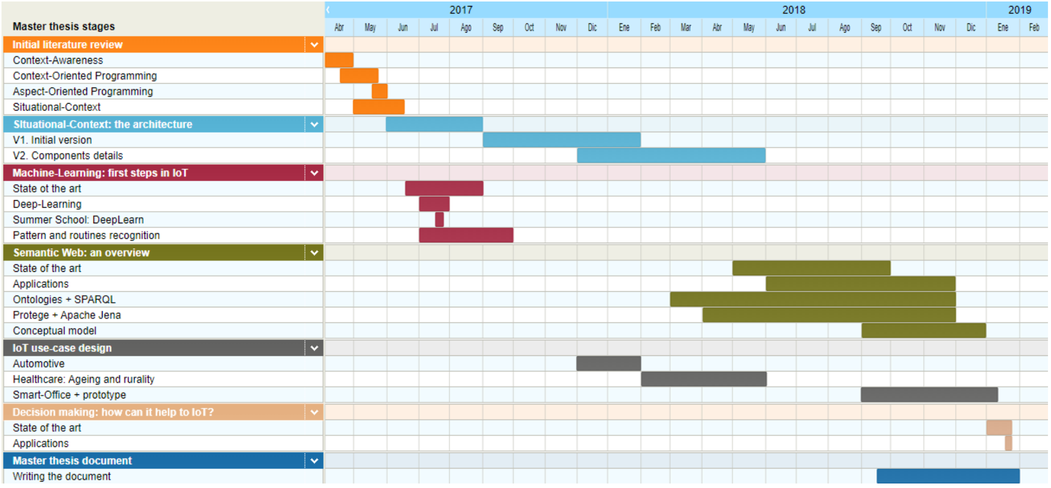


Figure 4.3: Master thesis task schedule

out and the problem to be solved is highlighted. A first version of the architecture that supports the paradigm is made along with the main components. Later these components were detailed and the data flow was defined. This architecture is subject to change, going through different revisions, and is currently still being perfected.

3. **Machine-Learning: first steps in IoT.** First contact with Machine-Learning technologies. This stage focuses on understanding the functioning of this technology as well as seeing its applications. Likewise, I also get into the Deep-Learning paradigm, as a requirement for the summer school I attend to lay down the basic knowledge. Emphasis is placed on two main types of applications that are of interest to our work: pattern recognition and routines.

4. **Semantic Web: an overview.** The need to relate devices is detected and semantic web is proposed as an interesting solution for this purpose. The first step is to review the state of the art to understand its operations and applications. Then, the use of ontologies as a method of relating devices, as well as reasoners and query languages. Tests are carried out with specific software and the conceptual model that integrates the semantic web with Situational-Context begins to be developed.

5. **IoT use-case design.** Several use cases are designed to show the problem detected, as well as the proposal designed. These use cases are framed in different IoT domains, to show the adaptability of the solution. Cases of use within the automotive sector, health care and intelligent office are proposed. This work only includes the last two.

6. **Decision making: how can it help to IoT?** Decision making is an important process in the adaptability of devices to people's needs. This stage is still in an initial state. Part of the state of the art and some of its applications and types of algorithms have been reviewed.

7. **Master thesis document.** The drafting of the document covers the last four months. During this time, the most important advances of the doctoral thesis are compiled and presented in an orderly manner in order to make the proposal known to the reader.

Chapter 5

Enabling IoT collaboration

This Chapter provides a solution to the three objectives proposed in this master thesis. Firstly, the problem of interoperability, produced mainly by endogamic practices carried out by manufacturers. Secondly, to adapt the devices behaviour to users preferences. And, finally, to improve users' privacy.

To achieve this, a general architecture is proposed that supports the Situational-Context paradigm. This paradigm covers many fields, such as establishing the connection through the devices within the same environment, discovering the relationship between goals and skills, capturing and inferring information from entities (devices and people), carrying out strategies that emerge from the different situations given in a context and providing entities the ability to learn from situations. In this master thesis, special attention will be paid in the relationship between goals and skills, proposing a specific architecture that covers this area. In this way, interoperability between devices is achieved. In addition, the development of a conceptual model, capable of modeling all the information of the situations as well as that of the users. This model has been developed for two purposes: first, to use the information contained in the user virtual profiles to adapt the behavior of the devices, and, second, to ensure the privacy of users by keeping all their information on their device and allowing them to share that which they want or need with other entities.

5.1 General architecture: Situational-Context

The architecture that supports the Situational-Context paradigm and models a basic entity has already been presented, as well as its components (Fig. 5.1). It is detailed below:

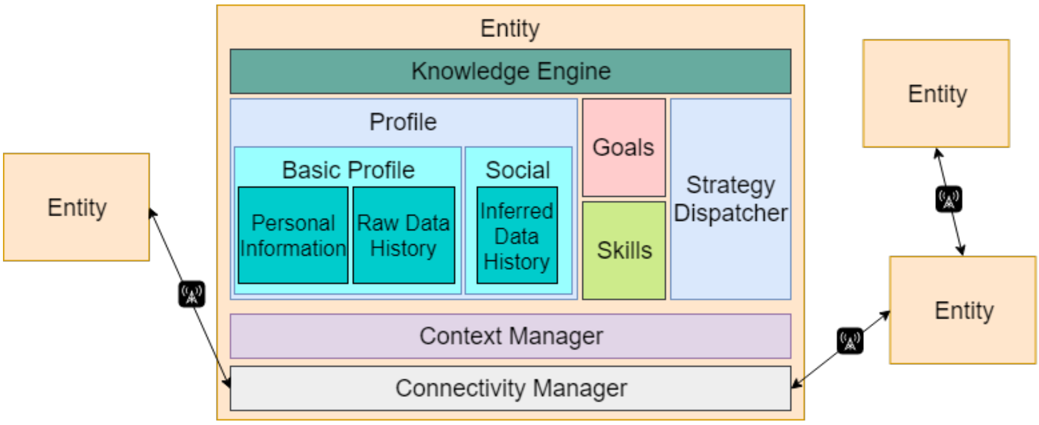


Figure 5.1: Entity architecture

- **Connectivity Manager.** Establishes the physical connection between entities. It sends and receives information related to skills, goals, personal information, etc.

- **Context Manager.** Responsible for creating and updating contextual information. It contains the information of the entities belonging to the same situation in a given instant of time.

- **Profile.** Union of the *Basic* and the *Social Profile* of the entity.

  - **Basic Profile.** Basic information that identifies the entity, such as the identifier, manufacturer, model, date of manufacture, etc. (Personal Information). It also contains raw data about the history of interactions with other entities (Raw Data History).
  - **Social Profile.** Stores all inferred data from the basic profile (Inferred Data History).

- **Skills.** Entity features. They produce a change in the context.

- **Goals.** They arise when an entity wants to obtain a state in a property of the environment that with the own capacities is not possible.

- **Strategy Dispatcher.** Devices can detect what goals there are in the environment, and which ones can be solved with their skills. A strategy is identified when it is detected how to coordinate the devices in the environment to solve the given goals. The complexity of strategies lies in the collaboration of entities to identify and solve needs.
- **Knowledge Engine.** Analyzes the history of the entity's activities to detect patterns and learn from them, with the goal of automating tasks in the future or detect routines.

This architecture achieves the interconnection of IoT devices at the features level. The interconnection is based on relating the skills of one entity with the goals of another. We know that the goals in an entity arise from the lack of skills when obtaining a desired state in the environment, so we must know how to perform this interconnection and that the goals can be resolved in the best way. Each entity has its own vision of the context, and knows the skills and goals of nearby entities, so that it can interact with them. This is achieved by integrating Situational-Context with Semantic Web and ontologies.

Entities virtual profiles are developed following the PeaaS paradigm [34] and using the novel tool developed by Tim Berners-Lee's team: Solid (Social Linked Data) [35]. Solid proposes a decentralized platform for social web applications, where user data is managed independently of the applications that consume this data, a proposal that is quite aligned with our work. Solid is a framework that can be used to implement the basic pillars of PeaaS, obtaining the following benefits [36]: (1) *True data ownership*, where users should have the freedom to choose where their data resides and who is allowed to access it; (2) *Modular design*, because applications are decoupled from the data they produce, users will be able to avoid vendor lock-in, seamlessly switching between apps and personal data storage servers; and (3)*Reusing existing data*, developers will be able to easily innovate by creating new apps or improving current apps, all while reusing existing data that was created by other apps. This could be a good approach to build part of Situational-Context paradigm and could come in handful of benefits.

5.2 Specific architecture: Goals and Skills

To show the potential of goals and skill within Situational-Context a specific architecture has been developed to model these components. Within a multi-device environment, the skills of one entity could cover the goals of another. Imagine a situation where there are several entities: both smart devices such as a media players, an air conditioning and a smart light; and several people. Each entity will have certain skills and goals associated with it. The relationship between skills and goals is given by the Semantic Web through ontologies, reasoners, and queries. Suppose that one person wants to listen to a certain type of music, while another needs to set a specific temperature. Nearby devices will detect these needs and use their skills to solve them. Likewise, two people want to establish a different illumination levels. In this case, the device with the skill to modify the room lighting will have to set a value that satisfies both of them. This case must also take into account if both people are equally important within the situation, if one is a guest, if the other commonly use that skill, or simply set an intermediate value for the situation. We are currently working on making decisions based on the hierarchy of users. The purpose of establishing some values or others will condition the level of comfort of these people.

Manufacturers usually offer an API to provide services and the features of their devices. Thanks to this API we can generate the different skills giving them a semantic connotation so that they can be easily interpreted and related to other skills or goals. Thus, the skills can be generated regardless of the technology used by the manufacturer and offered as services to other devices in the environment that need to use them. Fig. 5.2 shows the different components of the proposed architecture. In this way, features detected through the API of the media player, air conditioning and smart lights can be characterized within a common data model. In addition, the Semantic Web may provide additional information to enrich these devices, in order to learn more about them and exploit their full potential.

As shown in Fig. 5.2, an intermediate device is added to make the conversion among features and skills: a controller. The controller acts as a driver of the environment to make use of the API of each device. This controller is in charge of consulting and adding the smart devices located in the managed environment through a web interface. This interface makes it possible to determine which devices belong to the environment, adding new ones, deleting or editing them. Each environment can be delimited by a network or subnet, and all devices connected to it constitute the collaborative environment.

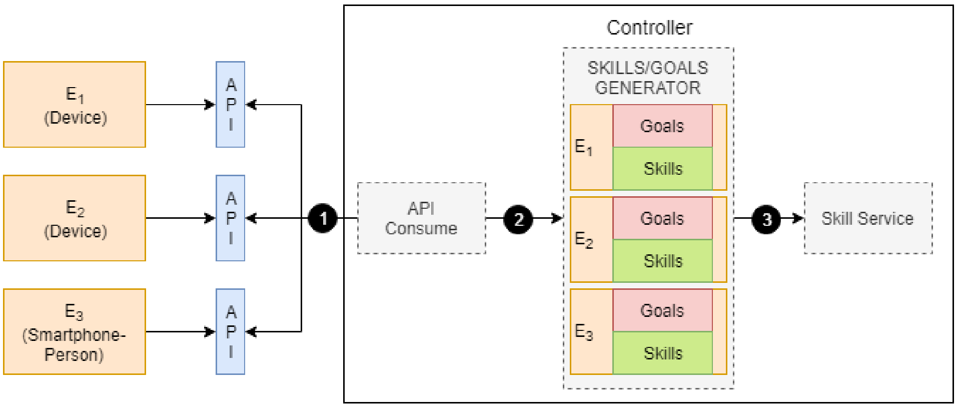


Figure 5.2: Architecture for skills and goals conversion

Therefore, the environment can be modified as necessary by using the web interface. The main features of this web interface are:

1. **Entity discovering:** The first requirement is to explore the environment regularly. On the one hand, when a new entity is discovered, for example a new illumination system, it can be included into the environment to check if it has some goals that need to be solved, or skills that can be used to cover other entities goals. But on the other hand, when a entity leaves the environment it has to be removed in order to maintain the environment goals and skills consistent, for example, when a person leaves the office. If a specific skill is covering a person goal when (s)he leaves the environment, its configuration must be re-adapted to the remaining entities within the environment.

2. **Entity management:** Skills are created from entity's features. The web interface allows us to explore the series of skills and specific goals defined for the discovered entity. As mentioned above, skills are created taking into account the API information from the entity, but the goals are defined manually. We are currently working on defining them from the entities' historical data, stored into the entity's virtual profile.

3. **Entity enrichment:** For each entity, we can get additionally knowledge checking external ontologies. This is because there are many data sources that, although they treat similar aspects for a certain smart device, each data source can follow its own specification to describe it. In order to obtain additional knowledge, an entity can be enriched with information from one or several remote ontologies.

5.3 Conceptual-Model

The conceptual model that supports the information contained in the entity's virtual profile is defined through ontologies. By using ontologies we can relate devices that have in common skills and goals, so that the goals of one entity can be covered by the skills of another. Goals are created from the processing of the characteristics of the environment. In other words, when a situation with certain characteristics is detected, these are associated with a specific goal and are stored in the virtual profile of the entity. This whole process is carried out at runtime.

Within this proposal, the use of ontologies has two purposes:

1. To serve each device has its own perception of the environment.
2. To provide additional information to the controller device, if it exists, to enriching the information of the devices in the environment.

The ontology of the conceptual model proposed here refers to the first of the two purposes described.

In this way, each entity will have a series of situations that it will have gone through over time. This information is stored in the virtual profile and it can be processed to infer and detect more precise situations. The way of sorting the situations that the entity is going through is called timeline. The situations of the timeline are generated from the features extracted from the situations themselves. Deep Learning techniques are used in this part (Feature Extraction). The information of these situations is valuable to be able to detect patterns of behavior or predict new situations in the future. This information is provided by the different sensors available to the devices such as GPS, lighting, humidity, accelerometer, etc. The conceptual model is detailed in Fig. 5.3.

When an entity detects other entities, the information stored in the profiles is exchanged. Each entity contains internally its ontological model with the data related to its profile and those around it. This data contains information about the skills, goals derived from situations through feature extraction techniques, situations timeline, personal information, etc. Even more complex skills and goals can be modeled where they can be composed by others.

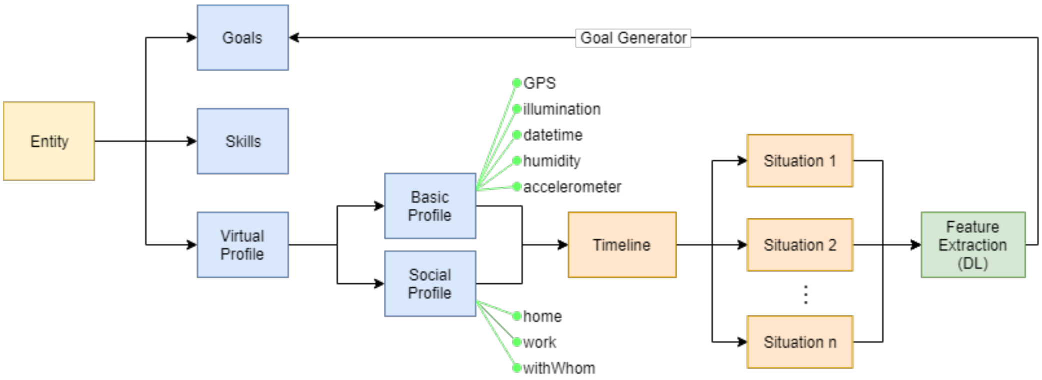


Figure 5.3: Conceptual model

In order to adequately explain this model, different aspects must be considered. These aspects are 1) the capture of entity data: since the vast majority of people have a smartphone, the data relating to where that person is, with whom, what time they get up, where they work, or anything that the device can detect, can be stored in the virtual profile of the user. Next, 2) the modeling of entities within the same context goes through the generation of an ontology. This ontology will be unique for each entity, and will contain contextual information about nearby entities, as well as their skills and goals. Once the entities have been identified, we move on to the next phase 3) of matching skills and goals. In this way we can cover the goals of one entity with the skills of another. The value or behavior of a skill, will go through 4) the interpretation of the data of the entities who need that skill to solve their goal and adapt their behavior as much as possible to their preferences. Finally, 5) possible patterns or routines will be detected for future situations that may happen, so that the adaptation to the context of WoT devices is more and more precise.

1. **Capturing entity data.** The capture of entity data involves the use of the smartphone, in case of people, or just the interaction with other entities, in case of smart devices. In case of people, using their smartphones, we can capture almost anything that happens around a person. With the passage of time, each person goes through different situations, situations that have valuable information, and that, in one way or another, must be stored to draw its potential. Talking about smart devices, data capture is carried out through the different sensors that the device itself may have, in the same way that is done with smartphones. All this information is stored in the virtual profile, represented through a database whose preliminary structure is detailed below in Fig. 5.4. Each entity stores its virtual profile and is only shared when and with whom it is de-

sired. For security purposes, this is an important aspect, as each entity can decide how to share its information.

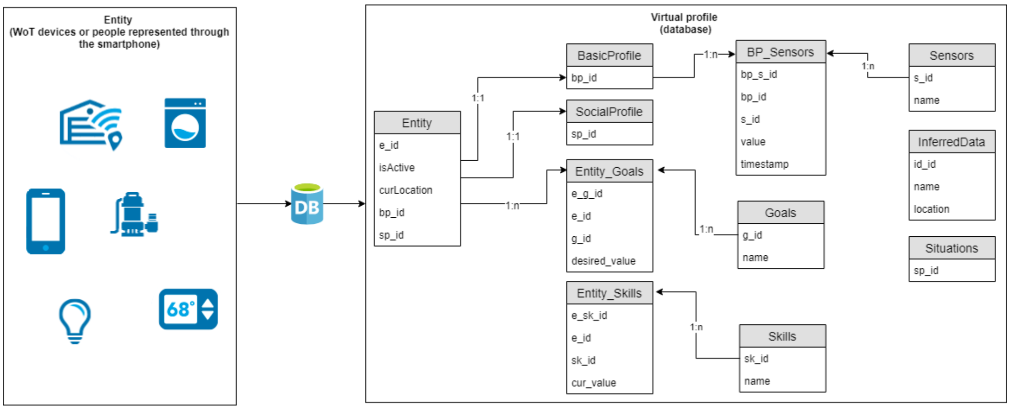


Figure 5.4: Virtual Profile

In order to not complicate the explanation of the virtual profile, the previous figure shows an outline of what would be the internal database that each entity would contain. The information in this database would be exchanged with the other entities in order to create a collaborative environment.

2. **Entity modeling.** The modeling of entities within a given context involves the use of ontologies and the Semantic Web, in order to obtain an ordered structure that is easy to interpret. In addition, with the Semantic Web we can increase data generation, access, flow, integration and comprehension using the same open standards that drive the World Wide Web (URI, HTTP and HTML), and that can be exploited to define, in the most faithful, way the environment surrounding a given entity. In this way, an entity knows that other entities are around it and navigate through their properties to determine if they can collaborate in any way in the resolution of needs. The proposed ontology is detailed below in Fig. 5.5.

3. **Matching skills and goals.** Once the ontology is defined, we can model all the entities present in the context through different ontological individuals. The resolution of the goals of a given entity is to find the skills that are capable of solving them adequately. This mapping is done through simple queries with SPARQL. The SPARQL ontology query language allows us, among other things, to identify which skill should be used to solve a given goal. This search is defined in a preliminary way through the name of the skill and goal. We are currently working

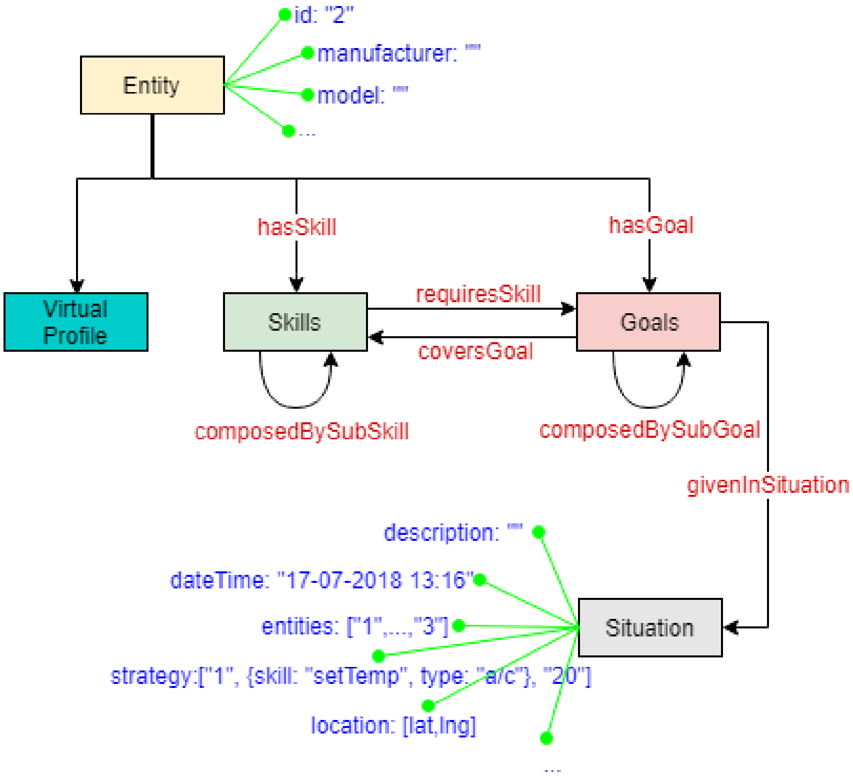


Figure 5.5: Ontology

on evolving the composition of these queries to make them more precise and complex. A simple example is shown below in Table 5.1.

In addition, thanks to the use of semantic reasoners we can obtain additional information about the entities. The reasoners will allow us to detect what types of entities we can find within the context, in order to more accurately search for skills and goals.

SPARQL queries give us the possibility to relate entities based on their skills and goals. At the same time, semantic reasoners provide us with a higher level of knowledge, as they can offer additional information to the one we are requesting. Both technologies are complementary, and depending on the complexity of the use case or the information we want to extract, we could use one or the other. The figure 5.6 shows how the use of a semantic reasoner like Hermit can extract extra knowledge from an ontology (Fig. 5.6a) and determine what type of individuals it possesses (Fig. 5.6b).

Thanks to these technologies we can relate one entity with another, from the information coming from its profiles. The tests carried out for the

Table 5.1: SPARQL query example

Query: context entities discovery

```
#context devices discovery
SELECT * WHERE
{
    ?Entity a :Entity .
}
```

Result

Entity
airconditioner
John
smartlights
Sara
...

Query: devices skills

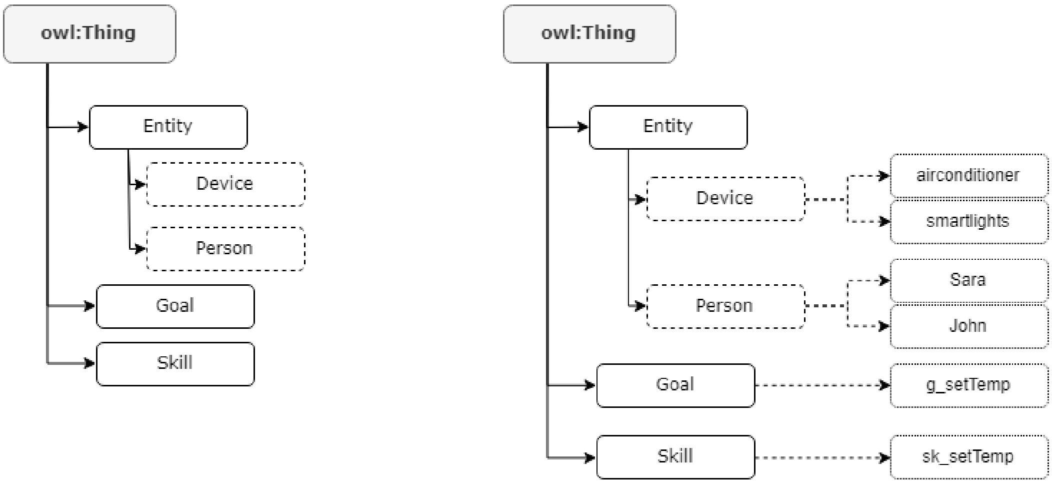
```
#show devices skills
SELECT * WHERE
{
    ?Device a :Device .
    ?Device :hasSkill ?skill .
}
```

Result

Entity	*Skill*
airconditioner	sk_setTemp
airconditioner	sk_setHumidity
smartlights	sk_setLuminosity
...	...

management of the virtual profile of the entities have been carried out with Apache Jena [37]. Jena is a framework for the Semantic Web based on Java that allows the treatment of ontologies. It was decided to use Jena because it has a version for Android (AndroJena), because the ontology processing would be performed on people's mobile devices and because it is one of the most widespread tools in the world of the Semantic Web, being also easily integrated with reasoners and SPARQL queries. Before that, preliminary examples were developed in Protégé, to check the ontology behaviour. Protégé is a free, open-source ontology editor and framework for building intelligent systems [38]. Once the example was studied, the next task was to use Jena to develop operations similar to those of Protégé, and to have more control over the proposed ontology.

4. **Adequacy of skills.** The skill that will solve a certain need has already been located. What value should it take? If it is used by several people, should it favor someone? This is a future line of work that is currently being deepened, like decision making or bio-inspired algorithms (Appendix A.4 and A.4.1). By using techniques like these, a skill will take the value calculated from the information stored in the virtual profile of the users in the context. The information that these algorithms will contemplate will be based on a hierarchical structure within the context, so that it can be decided whether to favour one person or to take an average value for all: which user is more important in this context? This section takes



(a) Initial ontology (b) Inferred ontology

Figure 5.6: Hermit reasoner usage

into account aspects such as the time a person spends in a given space (detecting whether they are at work, at home, etc.), who is around or whether they frequently use the skill requested.

5.4 Use-cases

During the development of this master thesis, the work has been carried out with different scenarios. An attempt has been made to cover a wide range of domains in order to have a diversity of information. Two use cases are shown below, one based on an Smart-Office, and the other based on health care, specifically for the elderly. We have also worked with another scenario based on an smart car, but we decided to not include it in this master thesis due to time availability.

5.4.1 Smart-Office

The following use case is based on an office. This use case is the one we have developed the most, and it uses the specific architecture shown above. This scenario has an intermediate device that acts as a controller of the environment. It has been decided to introduce this device in order to have a controlled environment and to facilitate the elaboration of the prototype, which is currently being developed. In Fig. 5.7 we show a use case based on an Smart-Office where all the elements of the architecture can be appreciated. These elements are the different entities within the environment as people and smart devices, a Raspberry Pi assuming the role of controller and a router to manage the network connectivity.

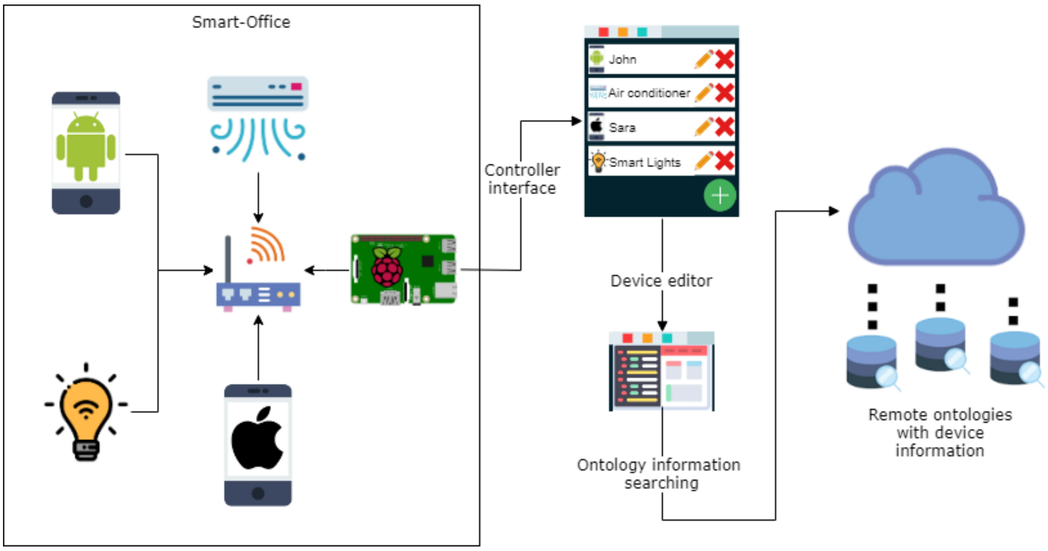


Figure 5.7: Smart-Office scenario

This proof of concept is focused on a Smart-Office. One day, a new Raspberry Pi is installed in the office and connected to the network. This device is able to manage the entities connected to the same network, both devices and people. The office has an smart air conditioning provided by Samsung that allows users to regulate the temperature. In addition, new smart illumination developed by Xiaomi was installed recently. Both smart devices are connected to the office router. The Raspberry Pi will be in charge of managing them, so it will act as a controller device for the office and, therefore, the environment. Therefore, the devices present in the network must be specified in this controller. In addition, when employees arrive to the office they must also be incorporated to the knowledge base of the controller so that it has evidence of them. The goals of the entities are specified manually through the controller's web interface, while the skills are built from the information in their APIs and incorporating extra information from external ontologies.

Therefore, the office is composed as follows:

- Four entities, two smart devices and two people.

Air conditioner: with the ability to regulate the temperature (*skill*).

Smart lights: lights that are able to adjust the illumination [1] (*skill*).

Sara: her virtual profile contains the following information:

- Desired temperature: 22.5º (*goal*)
- Ideal illumination: 6 (*goal*)

John: his virtual profile has the following data:

- Desired temperature: 20.5º (*goal*)
- Ideal illumination: 8 (*goal*)

- Two main devices to control the network and the rest of devices.

  Router: there is a router as a gateway to manage the network and to allow the communication among all devices connected to it.

  Raspberry Pi 3: the environment is controlled by a Raspberry Pi 3. This device is able to manage all devices skills and goals connected to the router and enrich them with external information.

Everyday, Sara arrives at the office at 8 in the morning. The controller detects her smartphone, an iPhone, and enables her profile automatically. The information related to her goals, such as the temperature and the illumination level, are recognized by the controller, because they already were introduced previously. Thanks to this information, the nearby devices into the network as the air conditioner and the smart lights can adapt their skills to her preferences.

An hour later, at 9 o'clock John arrives and his smartphone, an Android smartphone, is detected by the controller. On the same way, the controller enables his profile in order to know his goals and skills. Now, the controller must establish a strategy considering both people preferences. Smart devices must now satisfy the same goal for both of them. In this case, since the employees have the same seniority within the company and occupy the same professional category, an average value is taken to configure the skills that will meet their goals. Therefore, the air conditioner is set at 21.5º and the lights illuminate the room with a power of 7 out of 10.

Relating sklls and goals is done by the controller by using the information provided within the web interface. In Table 5.2 we show an example of the

[1] For this example we will use a scale for the luminous intensity from 0 to 10, where 0 is totally dark (without light) and 10 is totally illuminated.

SPARQL query to get this data. In this case, the air conditioner sets a temperature of 22.5° and the smart lights are configured to a luminous intensity of 6. In addition, entities can also learn from these situations. If a new entity appears, one of the current ones leaves the context, or produces a manual change on some of the skills, for example, manually setting the temperature to 20°.

Table 5.2: SPARQL query for ”temp” solution

Query: all present skills that cover the goal with ”temp” relation

```
#all present skills that cover the goal with ”temp” relation
SELECT * WHERE
{
?Goal a :Goal .
?Goal :requiresSkill ?skill .
?Goal rdfs:comment ?comment .
FILTER (CONTAINS(lcase(str(?Goal)), lcase(”temp”)))

?Entity a :Entity .
?Entity :hasSkill ?skill .
}
```

Result

Goal	g_setTemp
Skill	sk_setTemp
comment	Change temperature
Entity	airconditioner

This simple use case shows each and every one of the previously described modules, from the entities managed in the context, to adapt their skills to solve the detected goals. We can observe the technological interoperability issue through devices provided by different manufacturers like Samsung, Xiaomi, Apple and Android. Besides, several domains are shown if we consider that the air conditioner and the smart lights can easily belong to different domains, for example, industry and smart home. Even the situational problem is present when different people arrive or leave the office. The SPARQL query showed is just an example of all possibilities than may occur in a situation. We may use queries to detect devices, relationships, learn more about a goal or about a particular property.

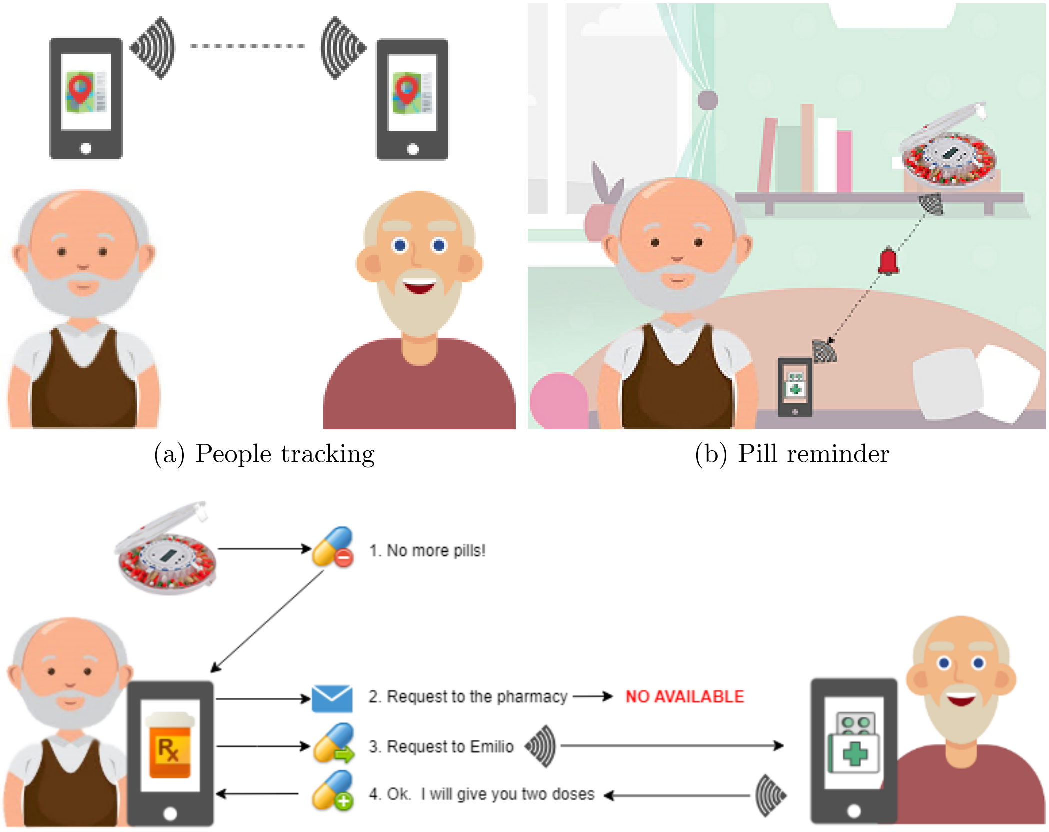
5.4.2 Ageing and rurality

The second use-case falls within the health care domain of IoT. To show the impact of the devices integration, we are going to use a scenario based on ageing and rurality. It is more abstract than the previous one use-case. Instead a controller, the entities are recognized each other within the environment. This scenario comes from a real use case of an European project where the author and the supervisors of this master thesis are involved: International Institute for Research and Innovation on Elderly (4IE) [39]. Its objective is to understand the biomedical, functional and psychological aspects of aging in specific contexts, generate new models and processes for elderly care and technological solutions that contribute to their health and quality of life and the sustainability of services.

Marty is a 74 years old man who lives in La Calera in the southeast of Cáceres, in the Las Villuercas Mountains. This morning, Marty went for a walk with his friend Rick. They both have their mobile phones with them. Since the village is running out of inhabitants and they are getting older, their sons and daughters always ask them not to go out without their mobile phones in case something happens to them. Marty and Rick are not aware of this, but their mobile phones can do much more than receive calls from their children. In addition, these devices are recording where they walk, where are them, with whom they are and are detecting each other so that their phones now know they are in company (Fig. 5.8a).

Marty has returned home and it's time to take his medicine. It is notified by the electronic pill dispenser that was given to him last month. This pill dispenser has also detected that there are no pills left for the next day (Fig. 5.8b). It is very important that Marty does not stop his heart treatment. Although his smartphone made the electronic prescription request, they were unable to bring his medication. There seems to have been a mistake at the pharmacy in Guadalupe, the only pharmacy nearby, and because it is a small town located in the rural area it is not possible for Marty to receive his medicine in time. Fortunately, Rick takes the same medication as Marty and received it last week. Marty and Rick have received a message telling them that tomorrow Rick must give Marty two doses (Fig. 5.8c).

The scenario presented above is subject to constant changes. We observe how the problem arises when a routine operation, such as ordering medication, can become complex and require additional operations due to problems that arise were not contemplated, such as the error in processing the electronic prescription or the lack of supplies from the pharmacy. Because of this, IoT devices must make decisions in real time to solve the needs of Marty and Rick. In this scenario there are a few changing conditions, but the possibilities that can occur are innumerable and cannot be defined at design time.



(a) People tracking

(b) Pill reminder

(c) Doses request flow

Figure 5.8: Ageing and rurality scenario

One of the main issues is that IoT devices must be configured manually by people, with the consequent investment of time and the need to have a minimal technical knowledge. But perhaps Marty and Rick do not have the necessary knowledge to be able to configure a smart device, or make it behave in a certain way and interact with other devices.

To mitigate this drawback, different works have promoted alternative methods to make IoT devices operate with each other, such as specific *frameworks*, e.g. [9], where a framework is developed to integrate specific domain applications into IoT, or [10], which presents interfaces and interconnection procedures based on oneM2M [40]. The use of ontologies and the Semantic Web are also becoming very important to solve these interconnection problems [41]. The Semantic Web and Internet of Things visions are converging toward the so-called Semantic Web of Things, that aims to enable smart semantic-enabled applications and services in ubiquitous contexts (SWoT) [42].

These works help to solve the problem of device interconnection, shown in the scenario presented before, but it is not an easy task, because technological diversity of smart devices must be taken into account. In addition, a correct handling of the context, where the own context conditions and people's needs are involved, must be taken into account. Unfortunately, this aspect hardly ever is consider. With Situational-Context we intend to do just that: on the one hand, to provide IoT devices with the capacity to detect at runtime characteristics of the environment such as temperature, humidity, location or detection of nearby devices and even people's needs; and, on the other hand, to provide the devices with the ability to evaluate the situation in order to make a decision that is capable of solving people's needs.

Using Situational-Context in the previously defined scenario it is possible to detect devices in the environment and their contextual information. In this case, Marty's pill dispenser realizes that there are no doses left, so it notifies Marty that he must place a new order through the electronic prescription. When Marty's smartphone tries to place the order, an error occurs which prevents him from being able to receive his medication. His smartphone automatically detects this situation, and as it knows that Rick is taking the same medication thanks to the information exchanged when they met previously, it proceeds to request a certain dose. Rick is warned of this, and his pill dispenser checks that he has an extra dose, so he can give some to Marty. In this way, an unforeseen situation produced by a constantly changing context is satisfactorily resolved at runtime. Identifying such situations at design time is almost impossible.

Chapter 6

Discussion

In this master's work we have approached the adaptation to the context of the Internet of Things devices, as well as their adaptation to the needs of people. Special attention must be paid to the problem of interoperability on a multi-device environment, and that are becoming increasingly important. Thanks to important works in this area we can detect the current state of device interoperability on the IoT and go a little further. These works make it possible to a greater or lesser extent to solve the problems of technological interoperability [43, 44], to favour communication among different domains [45, 46], to reduce the semantic distance between devices using the same technologies or belonging to the same domain [47], and even considering that in one scenario different situations may arise around smart devices [48].

We currently have a experimental version of the designed prototype. This prototype offers a fairly good result in controlled scenarios and is able to overcome the different dimensions detected. Regardless of the manufacturer, technology or IoT domain, any device can be added to the scenario though the web interface deployed in a Raspberry Pi. The use of the device's API provides the different features that are parsed to skills to be offered as a service later. In addition, each added device is enriched with information from external ontologies existing on the web, which allows us to increase the knowledge about it. Moreover, as the skills can behave differently depending on the situation, different situations have been tested for the same skills with a satisfactory result, specifically the variation of temperature depending on the number of people and the date and time within the environment.

In addition, thanks to the information shared through virtual profiles, entities can adapt their behavior depending on the context in which they find themselves. This is very useful for detecting specific situations or establishing specific behaviours according to circumstances.

Although the presented proposal offers many advantages, we are aware of its limitations. The relationship that is established between goals and skills is simple, and we are currently working on algorithms that contribute to compose strategies that help to satisfy one or several goals combining different skills (composed skills and goals), and to determine a specific order of action. In addition, the ontological repository that brings knowledge to the devices must be progressively nurtured in such a way that there is enough diversity to cover as many devices as possible, a fact that will require a lot of time and effort. Therefore, future work addresses issues of accessibility to the resources of the devices that manufacturers provide, the API, in terms of knowing and simplifying their discovery to be consumed. We are also working on performance, to limitate the resource consumption we are limiting situations to a specific network or subnet. Our efforts are also focused on exploring and knowing the different existing ontologies to enrich the knowledge about entities. Related to the skills configuration, we consider this aspect to be a major problem from the point of view of adaptation to users: what value should they take?; if it is used by several people, should it favor someone? This is a future work that is currently being deepened. By using techniques like decision making or bio-inspired algorithms, a skill take the value calculated from the information of the entities within the environment. The information that these algorithms contemplate are based on a hierarchical structure within the context, so that it can be decided whether to favour one person or to take an average value for all: what user is more important in this environment? This question takes into account aspects such as the time a person spends in a given space (detecting whether they are at work, at home, etc.), what people they have around them or whether they frequently use the skill requested.

Chapter 7

Conclusions and Future Work

Adaptation to the context of IoT devices is still a major problem. In addition, care must be taken to ensure effective collaboration between these devices so that they can meet people's needs.

This adaptation is not easy if we consider the large number of smart devices around us, that each of them follows the protocols and rules defined by a specific manufacturer and that, therefore, the collaboration between them becomes more and more complex.

We are aware that this is an ambitious project, as each objective detailed in the Chapter 2, may constitute a complex project by itself, but we believe that the contribution of this work is valuable from the point of view of specifying the problems detected, as well as a preliminary solution and even the prototype that is being developed.

This work has the backing of numerous scientific publications in conferences, so its validation has been done over several months. Although the Situational-Context paradigm still has a long way to go to consolidate itself as a real solution, both the work done so far, as well as the work under development and the one that is about to arrive, will make it easier for smart devices to meet people's needs.

Finally, on the one hand, the most relevant contributions of this work can be summarized as follows:

1. Achieve device interoperability:
    - Use of goals and skill to overcome the barrier of the different existing protocols and manufacturers.
    - Use of the semantic web to enrich the information of the devices of the environment.
2. Adaptation to the context of these devices to meet people's needs:
    - Depending on the users' preferences, by means of the information stored in the virtual profile.
    - By exchanging virtual profile information between entities or with the environment controller device.
    - Identification of situations where a goal can be covered by a certain skill.

On the other hand, some of the future lines that this work leaves open are:

1. Make decisions when establishing a specific value for a skill, using decision making or bio-inspired algorithms.
2. Detect situations where goals and composite skills can be found. (use several skills to solve a goal)
3. Elaborate an ontological repository that covers most of the possible devices to enrich the given environments.
4. Provide a learning capacity to the devices from the situations lived, with the aim of specifying as much as possible the use of a skill to cover a goal.

Chapter 8

Personal Reflection

As I said throughout this master thesis, this work is part of a more ambitious project that I am developing as a doctoral thesis: the paradigm of Situational-Context.

The development of this work has helped me to strengthen the most important ideas of the work carried out so far and to have a snapshot of the current state of the doctoral thesis. In addition, it has helped me to clarify the strengths and weaknesses that the work presents, and to be able to focus the efforts in the most necessary parts. It has also helped me to remember the most important literature that I reviewed several months ago, and which is still of great interest today.

This work at the end of the Master degree is no more than a small step to show the potential that the Situational-Context paradigm has in multi-device environments developed in the Internet of Things, where the discovery and collaboration of intelligent devices are crucial for their adaptation to the context and satisfaction of people's needs.

I think that in the near future, where there will be millions of smart devices connected to the Internet, we will need a tool that allows us to manage them, make the most of them in order to satisfy people's needs and understand the situations in which they find themselves. For this reason, the Situational-Context paradigm presents all the ingredients to achieve all of this.

Appendix A

Related Research Areas

This appendix contains those research areas that we consider relevant for this work.

A.1 Context-Oriented Programming

Within this master thesis, Context-Oriented Programming becomes important because of its relevance within context-oriented development. This is very well aligned with our goal of making IoT devices aware of the context they are in and acting accordingly.

Robert Hirschfeld, Pascal Costanza and Oscar Nierstrasz are considered the parents of Context-Oriented Programming [49].

As we saw at the beginning, contextual information plays an increasingly important role for applications and services ranging from those that are location-based to those that are situation dependent or even deeply personalized. While context-awareness is already an integral part of regular business applications, it is becoming critical for mobile and ubiquitous computing, where devices must adapt their behavior to the services available in their current environment.

Despite the fact that context is clearly a central notion to an emerging class of applications, there is little explicit support for context awareness in mainstream programming languages and runtime environments. This makes the development of these applications more complex than necessary. For that, Context-oriented Programming (COP), treats context explicitly and makes it accessible and manipulable by software.

One difficulty in proposing a concrete COP language is that context covers a wide range of concepts ranging from domain-specific to technology-dependent attributes, and including properties that may be spatial or temporal, or even based on hardware or software. We see personalization, sharing, location-awareness, ubiquitous computing, software evolution, runtime adaptation, and execution context dependencies as a part of the application domain of COP. The lack of programming mechanisms to support the development of context-aware applications forces the design of these applications to be more complex and fragile than necessary.

Therefore, COP is a programming technique to enable context-dependent computation. COP brings a similar degree of dynamicity to the notion of behavioral variations that object-oriented programming brought to ad-hoc polymorphism. Many researches in this field dedicated to realise COP paradigm by integrating COP constructs to an existing language. e.g ContextJ (COP + java), ContextL (COP + Lisp), ContextR (Ruby), ContextS, ContextLua, PyContext, etc [50]. The main characteristics of COP are:

- COP is independent from 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.

A.2 Ambient-Intelligent

Another objective is to make the integration of devices and people increasingly automatic. We consider Ambient-Intelligent as relevant to understand how to do this within a multi-device environment of the IoT, where in addition there are people who must make use of these devices.

Ambient Intelligence (AmI) is the vision that technology will become invisible, embedded in our natural surroundings, present whenever we need it, enabled by simple and effortless interactions, attuned to all our senses, adaptive to users and context and autonomously acting. High quality information and content must be available to any user, anywhere, at any time, and on any device [51].

There are many settings in which AmI can greatly impact our lives. Some of these applications have already been pursued by AmI researchers. Sample areas of AmI application have been extracted from [51].

- **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, Patient 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 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 enhance 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 work, which aims to achieve the resolution of people's needs through intelligent devices.

The following evolution on the behavior of the devices is given by machine learning technologies.

A.3 Machine-Learning

In addition, devices are becoming increasingly intelligent thanks to learning technologies. In this sense, Machine-Learning technologies are perfectly aligned with another of the objectives of this work, in which IoT devices must learn to detect specific situations or factors in order to automate their behavior.

Collaboration in a multi-device environment must have an important learning component in order to detect different situations and allow devices to adapt to them. Thanks to Machine-Learning technologies, devices can learn from people's routines or detect behavioural patterns. The following paragraphs are an extract of a basic guide in machine learning published in [2], and that it will clarify the operation of machine learning techniques.

Machine learning, as the name suggests, provides machines with the ability to learn autonomously based on experiences, observations and analysing patterns within a given data set without explicitly programming. When we write a program or a code for some specific purpose, we are actually writing a definite set of instructions which the machine will follow. Whereas in machine learning, we input a data set through which the machine will learn by identifying and analysing the patterns in the data set and learn to take decisions autonomously based on its observations and learnings from the dataset.

While learning about machine learning basics, one often confuses machine learning, Artificial Intelligence and Deep Learning. The Fig. A.1 clears the different types of learning.

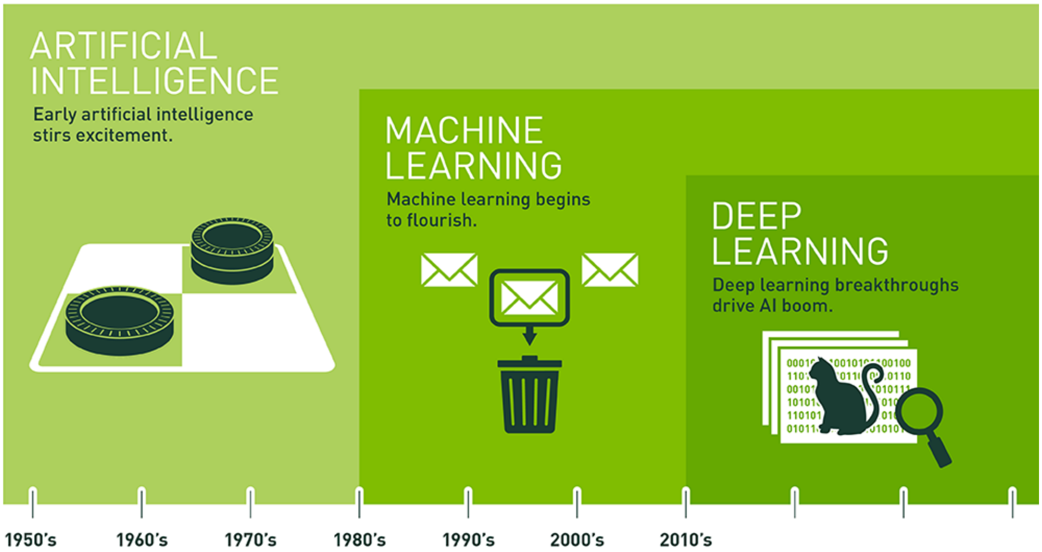

ARTIFICIAL INTELLIGENCE
Early artificial intelligence stirs excitement.
MACHINE LEARNING
Machine learning begins to flourish.
DEEP LEARNING
Deep learning breakthroughs drive AI boom.
1950's
1960's
1970's
1980's
1990's
2000's
2010's

Figure A.1: Machine learning evolution [2]

Depending on the type of learning applied, machine learning algorithms can be classified into three main categories:

1. **Supervised Algorithms:** There is a dataset on which the machine is trained. This is made up of tagged data or the required input and out-

put parameters. For example, classify whether a fruit is an orange or an apple. Here the orange and apple will be the labels and the training data set will already be classified into the given labels based on certain parameters through which the machine will learn these characteristics and patterns and will classify some new input data based on learning these training data. We can find several algorithms that perform this function, among them: linear regression, logistic regression, support sector machine (SVM), decision trees or random forest.

2. **Unsupervised Algorithms:** Unlike the supervised learning algorithms in which the dataset data is tagged, here the data that will teach the machine will not be tagged. The grouping of the data in a specific group will be done on the basis of the similarities between the variables or characteristics. The most popular algorithms within this domain are K-mean clustering and neural networks.

3. **Reinforcement Algorithm:** This type of machine learning requires to determine the ideal behaviour within a specific context, in order to maximize the rewards. It is based on the rewards and punishment principle which means that for any decision which a machine takes, it will be either be rewarded or punished due to which it will understand whether or not the decision was correct. In this way, the machine will learn to take the correct decisions to maximize the reward in the long run.

Machine learning has a myriad of applications and is being used in almost all the major fields. Similarly, machine learning has gained huge traction in the field of trading as well with domains such as Algorithmic Trading are witnessing exponential growth. Machine learning in trading is eventually automating the process of trading, wherein the machines themselves are becoming capable to learn from the previous data and take decisions to maximize profit or minimize loss. Fig. A.2 show some application domains of machine learning.

A survey by MIT and Google Cloud [52] demonstrates that 60% of the organisations have already been using machine learning strategies and one-third of them are at an early stage of development. This report by Forrester predicts huge growth for machine learning, which forecasts that the Predictive Analytics and Machine Learning (PAML) market will grow at 21% CAGR through 2021.

Therefore, the scope of this work includes Machine-Learning technologies to ensure that smart devices are able to learn from people to discover

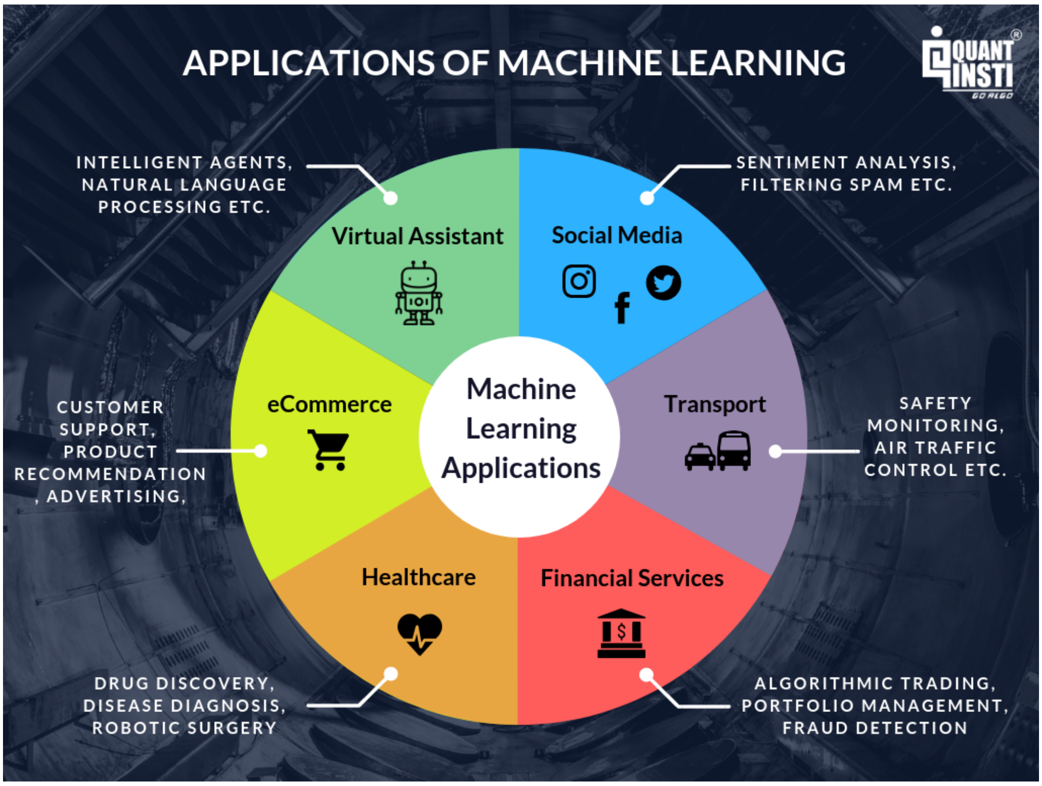

APPLICATIONS OF MACHINE LEARNING
QUANT INSTI
INTELLIGENT AGENTS, NATURAL LANGUAGE PROCESSING ETC.
SENTIMENT ANALYSIS, FILTERING SPAM ETC.
Virtual Assistant
Social Media
eCommerce
Machine Learning Applications
Transport
CUSTOMER SUPPORT, PRODUCT RECOMMENDATION , ADVERTISING,
SAFETY MONITORING, AIR TRAFFIC CONTROL ETC.
Healthcare
Financial Services
DRUG DISCOVERY, DISEASE DIAGNOSIS, ROBOTIC SURGERY
ALGORITHMIC TRADING, PORTFOLIO MANAGEMENT, FRAUD DETECTION

Figure A.2: Machine learning applications [2]

routines or different behaviors. In addition, they must detect or compose different situations where the smart devices involved in them are able to predict future behaviors and adapt as accurately as possible. The different types of learning detailed above constitute a fundamental point of this work. Although it remains to be decided which to use, it is clear that Machine-Learning technologies play a crucial point within a multi-device IoT environment.

A.4 Decision-Making

The change produced in a context is made to satisfy a specific need. This change must attend to people's preferences and should usually be a certain value such as a temperature, a song, a level of illumination, a notification or anything that an intelligent device can do. At this point, the next step is to determine that value: What temperature should be set? What music should be played? This value coincides with the information contained in the people's virtual profile. When it comes to a person, it's simple because you only have to establish that value. The real problem comes when there are more people involved in the change: what temperature to set for a room with several people? This is not easy to do and it is necessary to establish a decision-making process.

Decision making is usually defined as a mental process, which involves judging multiple options or alternatives, in order to select one, so as to best fulfil the aims or goals of the decision maker. Therefore, there are two main components involved in decision making: the set of alternatives, judged by the decision maker, and the goals to be satisfied with the choice of one alternative. The output of this process can be an action or an opinion of choice [53].

Decision making is a process. This means that in general it takes some time and effort until the choice is made, involving several activities [54, 55]:

- Identification of the decision problem
- Collecting and verifying relevant information
- Identifying decision alternatives
- Anticipating the consequences of decisions
- Making the decision
- Informing concerned people and public of the decision and rationale
- Implementing the selected alternative
- Evaluating the consequences of the decision

The key step of this process is making the decision automatically, that is, choosing the most preferred alternative using judgement based on available information. With the decision, we give precedence to the selected alternative, assuming (and hoping) that this alternative will provide the best (e.g., the easiest, most efficient, cheapest, safest, etc.) solution to our decision problem. The decision is considered a conscious and deliberate act, what makes the decision maker responsible for its consequences. The implementation of the decision often consumes resources, such as time, energy, money and willpower, and is therefore irrevocable. The consequences of a decision cannot be taken back; if necessary, they can only be affected by new decisions.

A.4.1 Bioinspired-algorithms

Bio-inspired algorithms are an interesting proposal for decision making within Situational-Context.

The domain of bio-inspired computing is gradually gaining prominence in current times. As organizations and societies move toward a digital age, there has been an explosion of data. This data explosion makes it increasingly difficult to extract meaningful information and gather knowledge using standard algorithms, due to the increasing complexity of analysis. Finding the best solution is increasingly difficult to identify, if not impossible, due to the breadth and dynamism of the solutions and the complexity of the calculations. Often, the optimal solution for such a difficult NP problem is a point in n-dimensional hyperspace and identifying the solution is computationally very expensive or even not feasible in a limited time. Therefore, intelligent approaches are needed to identify suitable working solutions.

In this context, intelligent meta-heuristics algorithms can learn and provide a suitable working solution to very complex problems. Within meta-heuristics, bio-inspired computing is gradually gaining importance as these algorithms are intelligent, can learn and adapt like biological organisms. These algorithms are attracting the attention of the scientific community due to the increasing complexity of the problems, the growing range of potential solutions in multidimensional hyperplanes, the dynamic nature of the problems and limitations, and the challenges of incomplete, probabilistic, and imperfect information for decision making [56].

Bibliography

[1] NCTA — The Internet Television Association. Infographic: The growth of the internet of things. https://www.ncta.com/whats-new/infographic-the-growth-of-the-internet-of-things. (Accessed on 2018-10-11).

[2] QuantInsti. Machine learning basics. https://www.quantinsti.com/blog/machine-learning-basics. (Accessed on 2018-11-19).

[3] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[4] Ding Yafei, Li Guanyu, and Zhang Hui. Semantic space-based semantic collaboration method in semantic web of things. *Computer Applications and Software*, 2:002, 2016.

[5] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4):2347–2376, 2015.

[6] Antero Taivalsaari and Tommi Mikkonen. A roadmap to the programmable world: software challenges in the iot era. *IEEE Software*, 34(1):72–80, 2017.

[7] Rodrigo Roman, Jianying Zhou, and Javier Lopez. On the features and challenges of security and privacy in distributed internet of things. *Computer Networks*, 57(10):2266–2279, 2013.

[8] Charith Perera, Arkady Zaslavsky, Peter Christen, and Dimitrios Georgakopoulos. Context aware computing for the internet of things: A survey. *IEEE Communications Surveys & Tutorials*, 16(1):414–454, 2014.

[9] Neela Shrestha, Sylvain Kubler, and Kary Främling. Standardized framework for integrating domain-specific applications into the iot. In *Future Internet of Things and Cloud (FiCloud), 2014 International Conference on*, pages 124–131. IEEE, 2014.

[10] Jaeho Kim, Jaeseok Yun, Sung-Chan Choi, Dale N Seed, Guang Lu, Martin Bauer, Adel Al-Hezmi, Konrad Campowsky, and Jaeseung Song. Standard-based iot platforms interworking: implementation, experiences, and lessons learned. *IEEE Communications Magazine*, 54(7):48–54, 2016.

[11] Payam Barnaghi, Wei Wang, Cory Henson, and Kerry Taylor. 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, 2012.

[12] Amy Nordrum. The internet of fewer things [news]. *IEEE Spectrum*, 53(10):12–13, 2016.

[13] Jun Zhou, Zhenfu Cao, Xiaolei Dong, and Athanasios V Vasilakos. Security and privacy for cloud-based iot: challenges. *IEEE Communications Magazine*, 55(1):26–33, 2017.

[14] Mung Chiang and Tao Zhang. Fog and iot: An overview of research opportunities. *IEEE Internet of Things Journal*, 3(6):854–864, 2016.

[15] Rob Van Kranenburg and Alex Bassi. Iot challenges. *Communications in Mobile Computing*, 1(1):9, 2012.

[16] E27. The advantages and disadvantages of internet of things. `https://e27.co/advantages-disadvantages-internet-things-20160615/`. (Accessed on 2018-10-10).

[17] RedAlkemi. Pros cons of internet of things. `https://www.redalkemi.com/blog/post/pros-cons-of-internet-of-things`. (Accessed on 2018-10-10).

[18] Techopedia. What does semantic web mean? `https://www.techopedia.com/definition/27961/semantic-web`. (Accessed on 2018-11-20).

[19] Matthias Klusch, Patrick Kapahnke, Stefan Schulte, Freddy Lecue, and Abraham Bernstein. Semantic web service search: a brief survey. *KI-Künstliche Intelligenz*, 30(2):139–147, 2016.

[20] Amelie Gyrard, Soumya Kanti Datta, Christian Bonnet, and Karima Boudaoud. 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, pages 336–341. IEEE, 2015.

[21] Michele Ruta, Floriano Scioscia, Giuseppe Loseto, and Eugenio Di Sciascio. A semantic-enabled social network of devices for building automation. *IEEE Transactions on Industrial Informatics*, 13(6):3379–3388, 2017.

[22] Floriano Scioscia, Michele Ruta, Giuseppe Loseto, Filippo Gramegna, Saverio Ieva, Agnese Pinto, and Eugenio Di Sciascio. A mobile matchmaker for the ubiquitous semantic web. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 10(4):77–100, 2014.

[23] Luiz H Nunes, Julio C Estrella, Charith Perera, Stephan Reiff-Marganiec, and Alexandre CB Delbem. The elimination-selection based algorithm for efficient resource discovery in internet of things environments. In *Consumer Communications & Networking Conference (CCNC), 2018 15th IEEE Annual*, pages 1–7. IEEE, 2018.

[24] Gregory D Abowd, Anind K Dey, Peter J Brown, Nigel Davies, Mark Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In *International symposium on handheld and ubiquitous computing*, pages 304–307. Springer, 1999.

[25] Pilar Castro Garrido, Guillermo Matas Miraz, Irene Luque Ruiz, and Miguel Ángel Gómez-Nieto. A model for the development of nfc context-awareness applications on internet of things. In *Near Field Communication (NFC), 2010 Second International Workshop on*, pages 9–14. IEEE, 2010.

[26] Qiang Wei and Zhi Jin. Service discovery for internet of things: a context-awareness perspective. In *Proceedings of the Fourth Asia-Pacific Symposium on Internetware*, page 25. ACM, 2012.

[27] Charith Perera, Arkady Zaslavsky, Peter Christen, and Dimitrios Georgakopoulos. Ca4iot: Context awareness for internet of things. In *Green Computing and Communications (GreenCom), 2012 IEEE International Conference on*, pages 775–782. IEEE, 2012.

[28] Javier Berrocal, Jose Garcia-Alonso, Carlos Canal, and Juan M Murillo. Situational-context: a unified view of everything involved at a particular situation. In *International Conference on Web Engineering*, pages 476–483. Springer, 2016.

[29] Joan Ernst Van Aken. Management research as a design science: Articulating the research products of mode 2 knowledge production in management. *British journal of management*, 16(1):19–36, 2005.

[30] Vijay Vaishnavi and William Kuechler. Design research in information systems. 2004.

[31] Paul Johannesson and Erik Perjons. *An introduction to design science.* Springer, 2014.

[32] Alan R Hevner. A three cycle view of design science research. *Scandinavian journal of information systems*, 19(2):4, 2007.

[33] Ángela Barriga Rodríguez. Situational context: Gestión distribuida de información contextual. Master's thesis, 2017.

[34] Joaquin Guillen, Javier Miranda, Javier Berrocal, Jose Garcia-Alonso, Juan Manuel Murillo, and Carlos Canal. People as a service: A mobile-centric model for providing collective sociological profiles. *IEEE software*, 31(2):48–53, 2014.

[35] Essam Mansour, Andrei Vlad Sambra, Sandro Hawke, Maged Zereba, Sarven Capadisli, Abdurrahman Ghanem, Ashraf Aboulnaga, and Tim Berners-Lee. A demonstration of the solid platform for social web applications. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 223–226. International World Wide Web Conferences Steering Committee, 2016.

[36] Tim Berners-Lee. Solid vision. `https://solid.mit.edu/`. (Accessed on 2018-10-09).

[37] Apache Foundation. free and open source java framework for building semantic web and linked data applications. `https://jena.apache.org/`. (Accessed on 2018-10-12).

[38] Stanford University. Protégé. `https://protege.stanford.edu/`. (Accessed on 2019-01-04).

[39] International institute for research and innovation on elderly. `http://4ie.spilab.es/`. (Accessed on 2018-10-12).

[40] Jorg Swetina, Guang Lu, Philip Jacobs, Francois Ennesser, and JaeSeung Song. Toward a standardized common m2m service layer platform: Introduction to onem2m. *IEEE Wireless Communications*, 21(3):20–26, 2014.

[41] Ioan Szilagyi and Patrice Wira. Ontologies and semantic web for the internet of things-a survey. In *Industrial Electronics Society, IECON 2016-42nd Annual Conference of the IEEE*, pages 6949–6954. IEEE, 2016.

[42] Antonio J Jara, Alex C Olivieri, Yann Bocchi, Markus Jung, Wolfgang Kastner, and Antonio F Skarmeta. Semantic web of things: an analysis of the application semantics for the iot moving towards the iot convergence. *International Journal of Web and Grid Services*, 10(2-3):244–272, 2014.

[43] Debasis Bandyopadhyay and Jaydip Sen. Internet of things: Applications and challenges in technology and standardization. *Wireless Personal Communications*, 58(1):49–69, 2011.

[44] Karen Rose, Scott Eldridge, and Lyman Chapin. The internet of things: An overview. *The Internet Society (ISOC)*, pages 1–50, 2015.

[45] Geng Wu, Shilpa Talwar, Kerstin Johnsson, Nageen Himayat, and Kevin D Johnson. M2m: From mobile to embedded internet. *IEEE Communications Magazine*, 49(4), 2011.

[46] Amelie Gyrard, Pankesh Patel, Soumya Kanti Datta, and Muhammad Intizar Ali. Semantic web meets internet of things and web of things: [2nd edition]. In *Proceedings of the 26th International Conference on World Wide Web Companion*, WWW '17 Companion, pages 917–920, Republic and Canton of Geneva, Switzerland, 2017. International World Wide Web Conferences Steering Committee.

[47] Jussi Kiljander, Alfredo D'elia, Francesco Morandi, Pasi Hyttinen, Janne Takalo-Mattila, Arto Ylisaukko-Oja, Juha-Pekka Soininen, and Tullio Salmon Cinotti. Semantic interoperability architecture for pervasive computing and internet of things. *IEEE access*, 2:856–873, 2014.

[48] Pieter Bonte, Femke Ongenae, Femke De Backere, Jeroen Schaballie, Dörthe Arndt, Stijn Verstichel, Erik Mannens, Rik Van de Walle, and Filip De Turck. The massif platform: a modular and semantic platform for the development of flexible iot services. *Knowledge and Information Systems*, 51(1):89–126, 2017.

[49] Robert Hirschfeld, Pascal Costanza, and Oscar Marius Nierstrasz. Context-oriented programming. *Journal of Object technology*, 7(3):125–151, 2008.

[50] QUORA. What is context-oriented programming? `https://www.quora.com/What-is-context-oriented-programming`. (Accessed on 2018-11-13).

[51] Diane J Cook, Juan C Augusto, and Vikramaditya R Jakkula. Ambient intelligence: Technologies, applications, and opportunities. *Pervasive and Mobile Computing*, 5(4):277–298, 2009.

[52] Machine Learning. The new proving ground for competitive advantage. Technical report, Technical report, Mit Technology Review, 2017.

[53] Marko Bohanec. Decision making: A computer-science and information-technology viewpoint. *Interdisciplinary Description of Complex Systems: INDECS*, 7(2):22–37, 2009.

[54] David S Skinner. *Introduction to decision analysis.* Probabilistic Gainesville, Fla, 1999.

[55] Robert T Clemen. Making hard decisions: an introduction to decision analysis. 1996. *Duxbury Press*, 1991.

[56] Arpan Kumar Kar. Bio inspired computing–a review of algorithms and scope of applications. *Expert Systems with Applications*, 59:20–32, 2016.