FAHED ALKHABBAS

TOWARDS EMERGENT CONFIGURATIONS IN THE INTERNET OF THINGS

TOWARDS EMERGENT CONFIGURATIONS IN THE
INTERNET OF THINGS

Malmö University,
Studies in Computer Science No 8,
Licentiate Thesis

© Fahed Alkhabbas, 2018
ISBN 978-91-7104-958-2 (print)
ISBN 978-91-7104-959-9 (pdf)
Holmbergs, Malmö 2018

FAHED ALKHABBAS
**TOWARDS EMERGENT
CONFIGURATIONS IN THE
INTERNET OF THINGS**

Malmö University, 2018
Faculty of Technology and Society
Department of Computer Science and Media Technology

Studies in Computer Science

Faculty of Technology and Society
Malmö University

1. Jevinger, Åse. Toward intelligent goods: characteristics, architectures and applications, 2014, Doctoral dissertation.
2. Dahlskog, Steve. Patterns and procedural content generation in digital games: automatic level generation for digital games using game design patterns, 2016, Doctoral dissertation.
3. Fabijan, Aleksander. Developing the right features: the role and impact of customer and product data in software product development, 2016, Licentiate thesis.
4. Paraschakis, Dimitris. Algorithmic and ethical aspects of recommender systems in e-commerce, 2018, Licentiate thesis.
5. Hajinasab, Banafsheh. A Dynamic Approach to Multi Agent Based Simulation in Urban Transportation Planning, 2018, Doctoral dissertation.
6. Fabijan, Aleksander. Data-Driven Software Development at Large Scale, 2018, Doctoral dissertation.
7. Bugeja, Joseph. Smart Connected Homes: Concepts, Risks, and Challenges, 2018, Licentiate thesis.
8. Alkhabbas, Fahed. Towards Emergent Configurations in the Internet of Things, 2018, Licentiate thesis.

Electronically available at:

<https://muep.mau.se/handle/2043/26522>

To those who never give up.

ABSTRACT

The Internet of Things (IoT) is a fast-spreading technology that enables new types of services in several domains, such as transportation, health, and building automation. To exploit the potential of the IoT effectively, several challenges have to be tackled including the following ones. First, the proposed IoT visions provide a fragmented picture, leading to a lack of consensus about IoT systems and their constituents. A second set of challenges concerns the environment of IoT systems that is often dynamic and uncertain, e.g. devices can appear and be discovered at runtime as well as become suddenly unavailable. Additionally, the involvement of human users complicates the scene as people's activities are not always predictable. The majority of existing approaches to engineer IoT systems rely on predefined processes to achieve users' goals. Consequently, such systems have significant shortcomings in coping with dynamic and uncertain environments.

To piece together the fragmented picture of IoT systems, we systematically identified their characteristics by analyzing and synthesizing existing taxonomies. To address the challenges related to the IoT environment and the involvement of human users, we used the concept of Emergent Configurations (ECs) to engineer IoT systems. An EC consists of a dynamic set of devices that cooperate temporarily to achieve a user goal. To realize this vision, we proposed novel approaches that enable users to achieve their goals by supporting the automated formation, enactment, and self-adaptation of IoT systems.

Keywords: Emergent Configurations, Internet of Things, Software Architectures, Self Adaptive Systems.

PUBLICATIONS

Included Papers

- Paper I. Alkhabbas, F., Spalazzese, R., Davidsson, P.: A Characterization of Internet of Things Systems through Taxonomies. Submitted to Journal.
- Paper II. Alkhabbas, F., Spalazzese, R., Davidsson, P.: Emergent Configurations in the Internet of Things as System of Systems. In: IEEE/ACM Joint 5th International Workshop on Software Engineering for Systems-of-Systems and 11th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems (JSOS), pp. 70-71. IEEE Computer Society, 2017. <https://doi.org/10.1109/JSOS.2017.6>
- Paper III. Alkhabbas, F., Spalazzese, R., Davidsson, P.: Architecting Emergent Configurations in the Internet of Things. In: IEEE International Conference on Software Architecture (ICSA), pp. 221-224. IEEE, 2017. <https://doi.org/10.1109/ICSA.2017.37>
- Paper IV. Alkhabbas, F., Ayyad, M., Mihailescu, R. C., Davidsson, P.: A Commitment-based Approach to Realize Emergent Configurations in the Internet of Things. In: IEEE International Conference on Software Architecture Workshops (ICSAW), pp. 88-91. IEEE, 2017. <https://doi.org/10.1109/ICSAW.2017.55>

- Paper V. Alkhabbas, F., De Sanctis, M., Spalazzese, R., Bucchiarone, A., Davidsson, P., Marconi, A.: Enacting Emergent Configurations in the IoT through Domain Objects. In: 16th International Conference on Service Oriented Computing (ICSOC), pp. 279-294. Springer, Cham, 2018. https://doi.org/10.1007/978-3-030-03596-9_19
- Paper VI. Alkhabbas, F., Spalazzese, R., Davidsson, P.: ECo-IoT: an Architectural Approach for Realizing Emergent Configurations in the Internet of Things. In: 12th European Conference on Software Architecture (ECSA), pp. 86-102. Springer, Cham, 2018. https://doi.org/10.1007/978-3-030-00761-4_6

Personal Contribution

For all publications above, except Paper V, the first author was the main contributor with regard to the planning and execution of the research as well as the writing of the publications. Regarding Paper V, the first three authors were the main contributors.

ACKNOWLEDGEMENTS

First of all, I would like to express my sincere gratitude and appreciation to Assoc. Prof. Romina Spalazzese, my amazing PhD supervisor, for her availability, patience, continuous support, and believing in me. This work would not have been completed without her efforts. It has been an honour to be her first PhD student. I also would like to express my special thanks and appreciation to my co-supervisor Prof. Paul Davidsson. I am deeply grateful for sharing your wealth of knowledge with me and for the inspiration, and great support you have been providing. Your guidance has been invaluable. Thank you both for your tremendous help and for everything. I could not have imagined having better supervisors for my PhD study.

I am so thankful to the “Internet of Things and People (IOTAP)” Research Profile financed by the Knowledge Foundation and Malmö University in collaboration with several industrial partners. I extend my thanks to all the participants in the research project “Emergent Configurations of Connected Systems (ECOS)”.

Many thanks to my PhD examiner Assoc. Prof. Andreas Jacobsson and to the review group members Prof. Bengt Nilsson and Dr. Radu-Casian Mihailescu for their efforts in reviewing this thesis and assessing my individual study plans. I would like also to convey my thanks to Assoc. Prof. Jan Persson, Dr. Bo Peterson, Dr. Christina Bjerkèn, Assoc. Prof. Johan Holmgren, Dr. Åse Jevinger, Dr. Annabella Loconsole, Prof. Thomas Pederson, Farid Naisan, Zahra Ghaffari, Sara Sätersten, Susanne Lundborg, and Solveig-Karin Erdal for the different types of support they have provided.

Furthermore, I would like to express my deep gratitude to my parents, wife, and children for their understanding, sacrifices, and support. Their contribution to every success I made in this life is uncountable. I am also thankful to my friends Majed Ayyad and Abed Alhakim Freihathat for their encouragement.

TABLE OF CONTENTS

I. COMPREHENSIVE SUMMARY	1
1. Introduction	3
1.1. Scenarios	4
1.2. Related Work	6
1.3. Research Questions	11
1.4. Outline	12
2. Research Methodology	15
2.1. Systematic Mapping Study	15
2.2. Design Science Research	16
3. Contributions	19
3.1. Characterization of IoT Systems	19
3.2. Emergent Configurations as Systems and SoS	20
3.3. Software Architectures for Emergent Configurations	21
3.4. Software Processes for Emergent Configurations	22
4. Conclusions and Future Work	25
4.1. Conclusions	25
4.2. Future Work	25
REFERENCES	29
 II. PAPERS	 37
PAPER I: A Characterization of Internet of Things Systems through Taxonomies	41
1. Introduction	42
2. Research Method	43
3. Results	48
4. Discussion	62

5.	Related Work	63
6.	Conclusions and Future Work	64
PAPER II: Emergent Configurations in the Internet of Things as Systems of Systems		
1.	Introduction	75
2.	ECs as Systems and SoS	76
3.	Conclusion	78
PAPER III: Architecting Emergent Configurations in the Internet of Things		
1.	Introduction	83
2.	The Smart Meeting Room Case	84
3.	Engineering Emergent Configurations	85
4.	Conclusion and Future Work	86
PAPER IV: A Commitment-based approach to realize Emergent Configurations in the Internet of Things		
1.	Introduction	97
2.	The Smart Meeting Room Scenario	98
3.	Conceptual Model for Commitment-based Emergent Configurations	99
4.	Realizing Emergent Configurations via Commitments . . .	101
5.	Conclusion and Future Work	105
PAPER V: Enacting Emergent Configurations in the IoT through Domain Objects		
1.	Introduction	109
2.	Background on the Domain Object Model	110
3.	The IoT-FED Approach	111
4.	The AL-office Scenario and AL Prototype Running on IoT-FED . . .	113
5.	Validation	117
6.	Lessons Learned	120
7.	Related Work	123
8.	Conclusion and Future Work	125
PAPER VI: ECo-IoT: an Architectural Approach for Realizing Emergent Configurations in the Internet of Things		
1.	Introduction	131
2.	Related Work	132
3.	Emergent Configurations Background	133
4.	The ECo-IoT Approach	135
5.	Prototype Implementation	136
6.	Experimenting ECo-IoT	141
7.	Conclusion and Future Work	144
		148

Part I.

COMPREHENSIVE SUMMARY

1. Introduction

The term “Internet of Things” (IoT) was first introduced by Kevin Ashton in 1999 [1]. The basic idea behind the IoT is to enable physical objects, or things, such as sensors, actuators, appliances, and vehicles, to connect and collaborate to achieve goals [2–4]. Exploiting such objects, the IoT has opened for new types of services that have a social and societal impact on almost every aspect of our daily life. New things are connected to the Internet every day and new types of applications are developed and used in several fields such as building automation, health-care, and transportation to mention a few [3, 5]. An example of a product that is available in the consumer market is Amazon Echo¹. It provides several services such as enabling users to control connected things like doors, TVs, lights, and thermostats through the Alexa² Voice Service. Another example is the Google Nest Thermostat³, which learns users behaviours to automatically adjust the usage of available power resources to save energy.

Connected things are often resource constrained with respect to energy sources and communication and processing capabilities [3, 6]. They can be equipped with sensors and/or actuators, can be autonomous, and are sometimes mobile. The number of things getting connected to the Internet is increasing rapidly and expected to reach 18 billion by 2020⁴. This expansion is due to several reasons including the advantages gained from connecting objects, the affordable technology, and that the connectivity is widespread.

To exploit the potential of the IoT effectively, a set of challenges have to be tackled including the following ones. First, the available IoT visions provide a fragmented picture, leading to a lack of common understanding about IoT systems and their constituents [3]. A second set of challenges concerns the environment of IoT systems that is often dynamic and uncertain e.g., things can appear and be discovered at runtime

¹<https://iot.do/devices/amazon-echo>

²<https://developer.amazon.com/alexa>

³<https://store.nest.com/product/thermostat/T3007ES>

⁴<https://www.ericsson.com/assets/local/mobility-report/documents/2017/ericsson-mobility-report-june-2017.pdf>

as well as become suddenly unavailable. The involvement of human users complicates the scene as people's activities are inherently situated and not always predictable. Related challenges have been addressed to different extents in several research disciplines such as Ubiquitous Computing [7, 8], System of Systems [9], Multi-Agent Systems [10], and Service Oriented Computing [11]. The majority of existing approaches for engineering IoT systems rely on predefined processes to achieve users' goals. Such systems have significant shortcomings in coping with dynamic and uncertain environments.

To piece together the fragmented picture of IoT systems, we systematically identified their characteristics by analyzing and synthesizing existing taxonomies. To address the challenges related to the IoT environment and the involvement of human users, we used the concept of Emergent Configurations (ECs) to engineer IoT systems. An EC is an IoT system composed of a dynamic set of things that connect and cooperate temporarily to achieve a user goal. A thing is any (smart) connected object, with its functionalities and services or applications [12, 13]. In the ECs vision, users express their goals without the need to worry about how available things can achieve them (if possible). To realize this vision, we proposed novel approaches that enable users to achieve their goals by supporting the automated formation, enactment, and self-adaptation of IoT systems. To better explain the concept of ECs, in Section 1.1, we introduce two concrete scenarios that we use throughout the thesis.

1.1. Scenarios

The concept of ECs represents a basis for novel approaches to engineer goal-driven IoT systems that can cope with dynamic and uncertain environments. Realizing the ECs vision requires enabling several processes including the automated interpretation of user goals, the dynamic discovery of available things, the automated formation and enactment of ECs to achieve the goals, as well as the automated adaptation of the ECs in response to changes in their environments. To better explain these concepts, in this section, we introduce two concrete scenarios that we use throughout the thesis. The first scenario, presented in Paper III, describes how an EC is realized to support a person who intends to give a presentation in a smart meeting room that she enters for the first time. The second scenario, presented in Paper V, illustrates how ECs are realized to enable a user to adjust light levels in arbitrary environments using an application.

1.1.1. The Smart Meeting Room Scenario

Lorraine enters an unknown smart meeting room and intends to give a presentation. The room is equipped with several things including light

and temperature sensors, curtains and light actuators, a smart screen, and a smart projector. She expresses her goal to give a presentation e.g., via an application installed on her smartphone. The goal is interpreted, the available things are automatically discovered, and an EC is automatically formed and enacted to achieve it. For instance, as illustrated in Figure 1a, the EC constituents could be the smartphone, the laptop, the smart projector, the light sensor, and the curtains actuator. The laptop connects and streams the presentation to the projector that illustrates it while curtains are closed automatically due to the high light levels detected by the light sensor. During the presentation, the projector turns off suddenly. The failure is detected and the EC is automatically adapted when Lorraine agrees to continue the presentation using the available smart screen as illustrated in Figure 1b. Since different display media require different light conditions, Lorraine is also proposed to open the curtains, when the presentation is resumed on the screen.

Figure 1.
The Smart Meeting Room Scenario

1.1.2. The Adjust Light Scenario

Adjust Light (AL)-office. Lorraine enters an office that is equipped with several connected things including light sensors and connected curtains and lights. The things, with their functionalities, are configured to be controllable by people in the office. Preparing for a meeting, she requests her goal to increase the light level using an application running on her smartphone. The available things are automatically discovered and, as illustrated in Figure 2, an EC formed by her smartphone, a light sensor, and connected curtain and light is enacted to achieve her goal. The light sensor measures the light level in the room, the light is turned on and the curtain is partially opened. We realized this scenario, including both its hardware and software, in our IoT lab.

AL-hotel. Lorraine enters a hotel room that is equipped with three light sensors, a connected curtain, and two connected lamps. The things, with

Figure 2.
The Adjust Light Scenario

their functionalities, are configured to be controllable by guests living in the room. Using the same application running on her smartphone, Lorraine expresses her goal to decrease the light level in the room. The available things are automatically discovered and given that the available curtain is closed, as illustrated in Figure 2, an EC formed by her smartphone, one of the light sensors, and the two lamps is enacted to reach her goal. The light sensor measures the light level in the room, one of the two lamps is turned off, and the light intensity level of the second lamp is reduced.

1.2. Related Work

In this section, we present a brief background about IoT systems and their constituents. Furthermore, we discuss related works in the fields of Software Architectures and Self-Adaptive Systems.

1.2.1. IoT Systems

There are four digitization waves that have made major impacts on our life. The first wave took place in 1980s when people started using computers to develop software products. The second wave happened a decade later when the Internet enabled computers to communicate, which helped people to share and access information easily. The third wave, mobile Internet, enabled us to exploit the Internet services wherever we are. Nowadays, we are witnessing the rise of the fourth digitization wave, referred to as the IoT, where the digitization is expanded to include physical objects such as sensors, actuators, vehicles, and appliances. These

objects, referred to as “Things”, represent basic building blocks of IoT systems [3, 4]. They add a physical dimension that makes IoT systems different from traditional software systems. Although in this thesis we do not focus on the hardware aspects of IoT systems, we take into consideration related aspects like things power source types, e.g., batteries, their connectivity status, i.e., connected/disconnected, and operational status, i.e., on/off.

As already mentioned, there is no consensus about IoT systems and their constituents. For instance, there are studies that considered smartphones, tablets and PCs as things [3, 14]; in contrast, other studies did not [15, 16]. There are studies that presented abstract characterizations of IoT systems [3, 17], and studies that provided more detailed characterizations but taking specific perspectives [6, 18, 19]. Multiple conceptual models for IoT systems have been proposed. Bauer et al. [20] presented a model for the IoT domain represented using the Unified Modelling Language (UML). Patel et al. [21] introduced a conceptual model for IoT applications. Ciccozzi et al. [22] proposed a conceptual model for mission-critical IoT systems by building upon the models presented in [20, 21]. However, none of these models presented a systematic characterization of IoT systems. In this section, we briefly introduce the main aspects about IoT systems while we refer the reader to Paper I for more details.

Figure 3.
A high level representation of an IoT system

Figure 3 illustrates a high level representation of an IoT system. It comprises a set of things that are heterogeneous, often resource constrained with respect to energy and processing capabilities, sometimes mobile, and can be autonomous. Connected things can be classified into three types:

1. Sensors or actuators: They provide the sensing or actuating func-

tions to monitor or change the state of the environment, respectively. They represent the majority of the connected things and have negligible or lightweight processing capabilities [6, 23];

2. Smart things: Physical objects are turned into smart things by being enriched with medium to high communication and processing capabilities that run smart software, they can also be equipped with sensors and/or actuators [23, 24];
3. Gateways: They are used to support connected things that do not possess the capabilities needed to individually connect, process requests, or handle security requirements. They represent the links between various involved things in an IoT system [24]. Their processing capabilities range between medium and high. Examples of gateways are Arduino⁵ and Raspberry Pi⁶.

Things connect, communicate and collaborate to achieve IoT systems goals [2, 4]. Communications should be enabled between users and things and among involved things. For this purpose, several types of networks can be used such as LAN, WLAN, WPAN [18, 25]. In addition, several protocols operating at different layers can be exploited such as MQTT⁷ and RPL⁸ [26]. The data generated by connected things and exchanged among them can be vast especially in large scale IoT systems. Such data is heterogeneous and can be of several types, for example, textual, numerical, images and videos [27]. IoT systems should be able to analyze such data and accordingly provide actionable insights in their environments [27, 28]. To facilitate the development of IoT systems, several IoT platforms have been provided such as AWS IoT Core⁹, Azure¹⁰, and Google Cloud IoT¹¹. They provide several services such as enabling things to communicate securely, monitoring their status, and processing the generated data.

As illustrated in Figure 3, the processing and storage capabilities of IoT system can be distributed across three layers namely, Cloud, Fog, and Edge [29]. In the Cloud Computing Paradigm, the capabilities of IoT systems reside in the cloud. Comparing to Cloud Computing, in the Fog Computing Paradigm, the capabilities reside in networks of more constrained devices (e.g., servers) that are closer to the edge of the network. Finally, in the Edge Computing Paradigm, processing capabilities reside in connected things [30, 31]. The Cloud Computing Paradigm has more powerful processing capabilities but also higher latency rates com-

⁵<https://www.arduino.cc/>

⁶<https://www.raspberrypi.org/>

⁷<http://mqtt.org/>

⁸<https://tools.ietf.org/html/rfc6550>

⁹<https://aws.amazon.com/iot-core/>

¹⁰<https://azure.microsoft.com/en-us/features/iot-accelerators/>

¹¹<https://cloud.google.com/solutions/iot/>

pared to the Fog and Edge paradigms. The Fog Computing Paradigm has less processing capabilities but also less latency rates compared to the Cloud Computing Paradigm. The Edge Computing Paradigm has the least latency rates but also has limited processing capabilities [30]. When deployed, IoT systems can be used by human or non-human users, i.e., (semi-) autonomous agents. [24].

1.2.2. Software Architectures

Software Architectures pervade and manifest themselves in all core aspects of software systems including their elements, i.e., components, interactions among them, and the constraints on the elements that enable systems to meet their design requirements [32–34]. Taylor et al. [35] highlight other important aspects in their definition about a system’s architecture, i.e., the important design decisions made during the system’s life cycle. In the literature, several methodologies have been proposed to design software architectures [36, 37]. The dynamicity and uncertainty of the IoT environment and the involvement of human users put additional requirements on engineering architectures for IoT systems. More specifically, such architectures should support the automated formation of goal-driven IoT systems that can self-adapt in response to changes in their dynamic and uncertain environments.

In the context of architectures, several works have been proposed. The IoT-A project presented a service-based reference architecture for the IoT [38]. Another novel reference architecture for enabling the notion of Plug and Produce in the industrial IoT is presented in [39]. Khan et al. [40] proposed a layered architecture for the IoT. Kramer et al. [41] introduced another architectural model to support the automatic (re)configuration of Self Managed Systems. The model exploits a repository of predefined plans to achieve goals. To realize new goals, new plans are generated in a time-consuming process.

Aura is an architectural framework that supports users to perform their tasks in mobile contexts by pre-compiling user tasks at design time and (re-)assigning appropriate services at runtime [42]. The Socrates Integration Architecture (SIA) is a service-oriented architecture that adopts a similar approach to enable the integration of the IoT in enterprise services [43]. Business processes are modelled at design time via an extended version of BPEL that allows the dynamic (re-)assignment of services at runtime. Systems realized by both Aura and SIA have limited flexibility as their execution processes are specified at design time. Dar et al. [44] proposed a high-level architecture to enable the adaptive composition of services in the IoT. The adaptation of composed services is performed at design time through user interfaces. Thus, automated adaptations in response to unforeseen changes are not supported.

In general, the majority of the existing architectures comprise repos-

itories responsible for providing systems execution flows, which are specified at design time, to achieve user goals. Therefore, they do not fully support the dynamic formation and enactment of IoT systems to achieve user goals in a specific context. Consequently, the systems that comprise them have shortcomings in adapting to unforeseen context changes.

1.2.3. Self-Adaptive Systems

To enable (IoT) systems to deal with continuously changing environments and evolving user goals, they are engineered as Self-Adaptive Systems [45, 46]. A well-established technique in the field of Self-Adaptive Systems is the implementation of feedback loops [47].

Several approaches have been proposed to engineer self-adaptive IoT systems. Hussein et al. [48] proposed a model-driven approach where an IoT system's states and adaptation triggers are modelled at design time. When an adaptation is triggered at the runtime, the system's state is changed based on the designed models. Ciortea et al. [49] proposed an agent-based approach for composing goal-driven IoT mashups where connected things are modelled at design time as agents or artefacts. Agents achieve their goals by executing predefined plans. In cases where their plans cannot achieve goals, agents interact and cooperate to compose a mashup that achieves them if possible. Seiger et al. [50] proposed a framework that implements the Monitor-Analyze-Plan-Execute plus Knowledge (MAPE-K) feedback loop for enabling the automated adaptation of workflow-based Cyber Physical Systems. Relying on predefined plans or workflows to achieve goals in [49, 50] limit the flexibility of IoT systems as it is hard to foresee the things that can be available in dynamic and uncertain environments. Likewise, relying on predefined models to enable IoT systems to self-adapt as in [48] limit the systems ability to respond to unforeseen changes in their environments.

The MobIoT is a service-oriented middleware that leverages an ontological based composition approach to achieve user requests by exploiting the notation of probabilistic registration and lookup mechanisms [51]. The approach enables the achievement of specific types of requests related to physics and chemistry domains. Mayer et al. [52] proposed an approach to enable the dynamic composition of service-based IoT mashups to achieve user goals. Connected things are modelled as semantic services and user goals are described in a machine understandable way. Given the description of a goal and a list of services, a service-based mashup is composed to achieve the goal if possible. The proposed approach supports also the automated adaptation of mashups apropos the dynamic availability of services.

In general, the majority of existing approaches rely on predefined processes to achieve user goals, thus limiting IoT systems flexibility. Therefore, there is a need for less-constrained approaches that enable the auto-

mated formation of IoT systems to achieve a user goals in a specific context. Such approaches should also support the automated enactment and adaptation of IoT systems in response to changes in their dynamic and uncertain environments.

1.3. Research Questions

In this thesis we address the following research questions:

RQ1. What are the characteristics needed to describe IoT systems?

As already mentioned, there is a lack of common understanding about IoT systems and their constituents. Therefore, the *objective* of RQ1 is to identify a set of characteristics that distinguish IoT systems from other systems. Such a set would capture the characteristics needed to be considered while engineering IoT systems. As presented in Section 1.2.1, several studies presented characteristics of IoT systems, however, there is no effort that presented a systematic characterization of IoT systems.

RQ2. What is a suitable software architecture to enable the dynamic formation and adaptation of goal-driven IoT systems?

The *objective* of RQ2 is to design a suitable software architecture that supports the automated formation, enactment and self-adaptation of goal-driven IoT systems in dynamic and uncertain environments. Engineering software architectures requires taking decisions about the system components, their responsibilities, relationships and the constraints on them [32, 33]. To design a suitable architecture, several challenges have to be addressed including specifying the relevant context dimensions and the components responsible for modelling and storing their values at the runtime. For instance, in the smart meeting room scenario, a context dimension that should be considered is the operational state, i.e., on/off, of the things. Moreover, proper relations and constraints must be specified among the components responsible for forming, enacting, and self-adapting goal-driven IoT systems. As presented in Section 1.2.2, several studies presented architectures for IoT systems, however, none of them can fully address the aforementioned challenges.

RQ3. How can goal-driven IoT systems be dynamically formed and adapted?

The *objective* of RQ3 is to devise processes that leverage the architecture, proposed in the answer to RQ2, to enable the automated formation, enactment and adaptation of goal-driven IoT systems in dynamic

and uncertain contexts. Considering the scenarios presented in Section 1.1, suitable architectures and processes should be designed to enable a user to adjust light levels in arbitrary environments using an application and to continue her presentation via the available smart screen after the sudden failure of the smart projector.

Devising such processes require addressing several challenges including proposing mechanisms to analyze how available things can collaborate to achieve user goals (if possible). For instance, how is it determined that the user's smartphone should connect and stream the presentation to the projector that then illustrates it? Furthermore, techniques are needed to maintain systems context at runtime and to detect events that affect the achievement of user goals. For instance, how can we maintain the operational state of the smart projector at runtime and detect that it has stopped working?

Moreover, mechanisms are needed to enable IoT systems to self-adapt in response to the detected events. For instance, how is it automatically determined that the user can continue the presentation via the available smart screen? As presented in Section 1.2.3, several approaches are proposed in related works. However, none of them can fully address the highlighted challenges.

1.4. Outline

This thesis is in the context and contributes to the project Emergent Configurations of Connected Systems (ECOS)¹² within the Internet of Things and People (IOTAP)¹³ Research Profile. The thesis is divided into two parts, the first part describes a comprehensive summary of it while the second part presents six papers that form the actual research of this thesis. Paper I is submitted to journal and is under review, while the other papers are accepted in peer-reviewed conferences and workshops. Figure 4 illustrates the relationships between these papers including their contributions and the research questions they address.

To answer to RQ1, in Paper I, we proposed a characterization of IoT systems by analyzing and synthesizing existing taxonomies, details about this contribution can be found in Paper I. Additionally, in Paper II, we studied ECs taking a System of Systems (SoS) perspective. This helped us better understand ECs, their characteristics, constituents, and their physical and logical layers. Thus, Paper II contributes also to answering to RQ2.

To answer to RQ2 and RQ3, in Paper III, we proposed an abstract architectural approach, which comprises an architecture and a process, for realizing ECs. Then, we proposed three novel approaches that re-

¹²<http://iotap.mau.se/projects/ecos/>

¹³<https://iotap.mau.se/>

Figure 4.
The relationships between the included papers and their contributions

fine it. They all enable users to achieve their goals by supporting the automated formation, enactment and adaptation of ECs. However, they exploit different techniques and focus on different aspects as follows. One approach exploits techniques adopted from the Self-Adaptive Systems and Artificial Intelligence domains to enable things to engage in agreements autonomously to form and adapt ECs. This approach refines the abstract process proposed in Paper III, details about this contribution can be found in Paper IV. The IoT-FED approach, proposed in Paper V, exploits the Domain Objects (DOs) technology to enable users to achieve their goals by supporting the automated formation of ECs as well as their enactment. It refines the abstract architecture and process presented in Paper III [11]. Finally, the ECo-IoT approach, proposed in Paper VI, exploits techniques adopted from the Self-Adaptive Systems and Artificial Intelligence domains to enable the automated formation and self-adaptation of ECs. It refines the abstract architecture and process presented in Paper III.

The remainder of Part I is as follows. Chapter 2 describes the research methodology applied to address the research questions. Chapter 3 presents the main contributions to the body of knowledge. Chapter 4 concludes the thesis and outlines possible future work directions.

2. Research Methodology

This chapter introduces the research methodology applied to address the research questions presented in Chapter 1. We applied a mixed research method that includes the Systematic Mapping Study (SMS) and the Design Science Research (DSR) presented in Sections 2.1 and 2.2, respectively [53–55]. We considered the peer reviewed literature as the main source of knowledge.

2.1. Systematic Mapping Study

To address RQ1, in Paper I, we conducted a SMS—a well-known methodology in the Software Engineering community [54]. The study identified the characteristics of IoT systems by analyzing and synthesizing existing taxonomies. In general, taxonomies are considered as reference models that facilitate the engineering and development of (software) systems [56]. The study followed the most commonly applied guidelines for conducting systematic mapping studies [54]. More specifically, the applied method involved three stages: *planning*, *conducting*, and *reporting*.

In the planning stage, we produced a well-defined review protocol. For this purpose, we carried out several activities including: confirming the need of the study, formulating the study goal, defining the RQs, formulating the search string, and generating selection criteria. Moreover, mechanisms were developed to analyze and synthesize the identified taxonomic dimensions. In the conducting stage, we set the developed protocol in practice, and we reported the results of the study in the reporting stage. Furthermore, we performed quantitative analysis to provide insights about the direction of the future research concerning IoT systems.

Critical Literature Review. Systems of Systems (SoS) and IoT systems / ECs have many common characteristics. For example, their constituents are managerially independent, i.e., self-controlled and managed, operational independent, i.e., can keep achieving their goals independently, heterogeneous, often distributed, and able to communicate. Also, their environments are often dynamic and uncertain [3, 57–59]. To better un-

derstand ECs, their characteristics, and constituents, in Paper II, we conducted an exploratory study that investigates ECs taking a SoS perspective by exploiting two scenarios. In the first scenario, an EC is formed by connected things to achieve a user goal while, in the second scenario, an EC constituent includes also two IoT systems.

2.2. Design Science Research

We applied the design science research to address RQ2 and RQ3 in an iterative way to devise approaches that realize the ECs vision. We chose this research method because we aimed to create innovative artifacts to solve practical problems—and this is supported by the methodology. The applied method comprises five stages: *problem explication*, *requirements definition*, *artifact design and development*, *artifact demonstration*, and *artifact evaluation* [55].

In the problem explication stage, in Papers III, IV, V, and VI, we explored different aspects of the problem by studying the literature, including examples of IoT systems, and evaluating related works. By leveraging the knowledge gained from answering the RQ1 and the contributions of Paper II, we identified the characteristics of the IoT systems addressed in this thesis (i.e., ECs). In the requirement definition stage, we formulated the functional requirements that ECs should meet.

In the artifact design and development stage, in Paper III, we proposed an abstract architectural approach for realizing ECs. Then, we proposed three approaches that refine it as follows. In Paper IV, we proposed an approach that enables connected things to engage into agreements autonomously to form and adapt ECs. In Paper V, we proposed the IoT-FED approach to enable the automated formation and enactment of ECs. In Paper VI, we proposed the ECo-IoT approach for enabling the automated formation and self-adaptation of ECs.

For all the approaches, the set of input parameters to the formation process comprises a user goal, e.g., give a presentation, available things with their capabilities, e.g., a light sensor measures light levels, and other contextual information such as things connectivity and operational status, e.g., on/off. The output of the formation process is an EC instance that achieves the user goal when enacted through the enactment process. The adaptation process requires additional parameters that represent the execution states of the ECs to be adapted.

In the artifact demonstration stage, we developed two prototypes that realized the adjust light and the smart meeting room scenarios presented in Chapter 1.1, these prototypes are described in Paper V and VI, respectively. In the evaluation stage, in Paper V, we conducted experiments to: (i) evaluate the ability of the IoT-FED approach to dynamically form different ECs based on available things; (ii) gain insights on

the IoT-FED performance. In Paper VI, we conducted experiments to get initial insights on the performance of some key components in the ECo-IoT approach. We plan to extend the approach presented in Paper IV, develop a prototype, and conduct experiments to evaluate it. We also plan to perform more extensive evaluations in our future work.

3. Contributions

This chapter presents the contributions of the thesis that were developed under the following assumptions:

1. The user goal specification and interpretation process is already performed and correctly terminated: Atomic goals resulting from decomposing a (high level) user goal, e.g., give presentation, into sub-goals are provided. Atomic goals are fine-grained goals achievable by things, for example, get light level is achievable by a light sensor. We plan to devise approaches for user goals interpretation in future work;
2. ECs are formed, enacted, and adapted within well-defined spatial boundaries: ECs goals are achieved within locations e.g., room, building. Consequently, the number of things involved in forming and adapting an EC is not expected to be massive. This notably mitigates the well-known IoT scalability problem [3].
3. ECs are realized to achieve goals within non-critical time constraints: We envision that ECs are realized within the timescale of seconds;
4. ECs are formed and enacted at runtime: We do not address cases where the formation and enactment of ECs are scheduled in the future;
5. Services that provide information about connected things and their locations are provided: We do not develop solutions to report information about connected things, for example, battery levels (when applicable), their connectivity status (i.e., connected/disconnected), operational status (i.e., on/off), and their locations.

3.1. Characterization of IoT Systems

The existing visions provide a fragmented picture about the IoT, leading to a lack of common understanding about IoT systems and their constituents. To fill this gap, in Paper I, we provided a holistic view of IoT systems. More specifically, we identified seventeen characteristics of IoT systems by conducting a SMS that leveraged existing taxonomies. We organized the characteristics into two categories: *elements* and *quality*

aspects of IoT systems, we refer the reader to Sections 3.1 and 3.2 in Paper I for more details about them.

The identified elements of IoT systems are IoT thing, communication, middleware, data, deployment, goal, user, identifier, resource management, and collaboration. The identified quality aspects of IoT systems are security, privacy, trust, interoperability, scalability, latency, and reliability. They represent non-functional characteristics of IoT systems. These characteristics represent the answer to RQ1. Moreover, we analyzed more in depth the elements of IoT systems and identified their dimensions. For example, the dimensions of the element IoT thing include, but not limited to, type and level of autonomy. The former dimension classifies connected things into the three types presented in Chapter 1.2.1, while the latter dimension classifies things based on their level of autonomy to non-autonomous and autonomous things.

Furthermore, by quantitatively analyzing the results of the SMS, we presented insights about the directions of future research concerning IoT systems. The analysis show that many taxonomies focus on security aspects of IoT systems while the most explored elements of IoT systems are IoT thing, communication and data respectively. We believe that the focus on securing IoT systems might be due to the sensitive data that they might monitor such as health parameters and the critical or costly control operations they might perform such as in self-driving cars and turning on district heating, respectively. The focus on the IoT thing element could be due to the physical dimension it adds to IoT systems, which makes them different from traditional software systems. Enabling IoT things to communicate and managing the generated data represent core aspects in the IoT [3, 5, 17, 60]; this might clarify the focus on the elements communication and data. The analysis also revealed that some elements of IoT systems such as user, goal, and deployment have not been explored sufficiently. They represent core characteristics of IoT system and we believe that they should be more investigated. By exploiting the identified characteristics, we provided a more formal characterization of IoT systems and also outlined a process for designing them. Further details about these contributions can be found in Paper I.

The proposed characterization represents a reference model that supports researchers and interested practitioners to (i) contribute to the research about IoT systems; (ii) better understand characteristics of IoT systems and take them into consideration when designing and developing IoT systems.

3.2. Emergent Configurations as Systems and SoS

In Paper II, we highlighted some common characteristics between the IoT and SoS domains [3, 57–59]. For instance, their constituents are het-

erogeneous, can be autonomous, able to communicate, often distributed, managerially independent (i.e., self-controlled and managed), and operationally independent (i.e., can keep achieving their goals independently). In addition, their environments are dynamic and uncertain. Therefore, we studied ECs taking a SoS perspective, which helped us better understand ECs, their characteristics, constituents, and their logical and physical layers. This contributed to answering to both RQ1 and RQ2.

The main outcome of this study is that ECs can be regarded as both systems and SoS based on the types of their constituents. More specifically, ECs are considered systems when their constituents are connected things and SoS when their constituents involve at least two IoT systems. Consequently, we believe that the IoT research can gain insights from the already developed SoS research and, vice versa, the IoT research might produce useful results for the SoS community.

3.3. Software Architectures for Emergent Configurations

To address RQ2, in Paper III, we proposed an abstract architecture for realizing ECs leveraging the characteristics of ECs presented in Chapter 1. As illustrated in Figure 5, the proposed architecture comprises the following components: (i) The *User Agent* (UA) is an application running on one of the existing smart things (e.g., smartphone) and used to enable users to interact with the system (e.g., to express her goal); (ii) the *Emergent Configuration Manager* (ECM) is responsible for automatically forming and enacting ECs to achieve user goals (if possible) and for adapting ECs in response to dynamic changes in the context; (iii) The *Thing Manager* (TM) is responsible for discovering and registering available things and services, monitoring their status and reporting any

Figure 5.
 A high level architecture for realizing ECs

changes to the ECM. Finally, a *set of connected things* that, following indications of the ECM, communicate and collaborate to realize ECs. The ECM comprises the following sub-components:

- (i) *Goal Manager*: It is responsible for interpreting user goals and forming ECs that achieve them when enacted;
- (ii) *Adaptation Manager*: It is responsible for adapting ECs in response to context changes. ECs are adapted by executing the Monitor-Analyze-Plan-Execute plus Knowledge (MAPE-K) loop -better described in Section 3.4 [61];
- (iii) *Context Manager*: It is responsible for maintaining ECs context;
- (iv) *Enactment Engine*: It is responsible for enacting ECs by commanding or requesting ECs constituents to perform functionalities in specific orders;
- (v) *System Knowledge Base (KB)*: It is the container of the context of ECs.

Papers V and VI present two possible refinements of the abstract architecture proposed in Paper III. In Paper V, we proposed the IoT-FED architectural approach for enabling the automated formation and enactment of ECs. It exploits heterogeneous and independently developed things, services, and applications modeled as Domain Objects (DOs), a service-based formalism [11]. The architecture refines the Thing Manager and the sub-components of the ECM by leveraging an IoT platform, the existing components of the DOs technologies, and also developing new ones.

In Paper VI, we proposed the ECo-IoT architectural approach that comprises another refinement of the abstract architecture proposed in Paper III. The architecture is designed to support the automated formation and self-adaptation of ECs. It refines the sub-components of the ECM by exploiting techniques adopted from the Self-Adaptive Systems and Artificial Intelligence domains [61–63].

Compared to the architectures of IoT systems presented in Chapter 1.2.2, the architectures we proposed answer better to the needs of IoT systems as they are designed to support the dynamic formation, enactment and self-adaptation of goal-driven IoT systems that can better cope with dynamic and uncertain environments.

3.4. Software Processes for Emergent Configurations

To answer RQ3, in Paper III, we proposed an abstract process for realizing ECs. In Paper V, we proposed the IoT-FED architectural approach for enabling the automated formation and enactment of ECs. As already mentioned, in this approach we model connected things, services and

applications as Domain Objects (DOs). The approach comprises a process that refines the abstract process presented in Paper III. The process starts by specifying a user goal type, e.g., adjust light level, and the goal spatial boundaries, e.g., Lorraine’s office. Given these inputs and using the concept of refinement supported by the DOs technologies, an EC is formed to achieve the user goal (if possible). The formed EC is then enacted by instantiating the DOs forming it.

To validate the feasibility of the IoT-FED approach, we developed a prototype that realized the adjust light scenario presented in Chapter 1.1 and performed experiments that revealed the IoT-FED ability to dynamically form different ECs when the number and/or the type of available things change. Also, the experiments showed that the IoT-FED approach enables the automated formation and enactment of the generated ECs in average time less than 2.5 seconds.

In Paper VI, we proposed the ECo-IoT architectural approach to enable the automated formation and adaptation of ECs. The approach comprises a process which refines the abstract process proposed in Paper III. Given the user goal type, e.g., give a presentation, and the goal spatial boundaries, e.g., a smart meeting room, an EC is formed from available things to achieve the goal (if possible). To enable the automated adaptation of the EC, the process implements the MAPE-K loop, a well-established technique in the domain of Self-Adaptive Systems [61]. The EC context is continuously monitored and changes are automatically detected and analyzed. The process exploits the notion of events to model changes in the context. In cases where the detected changes affect the achievement of the EC goal, plans are automatically generated, the EC is adapted accordingly and enacted, and the system KB is enriched.

We conducted an initial validation of the ECo-IoT approach by developing a first prototype that realized the smart meeting room scenario presented in Chapter 1.1. Moreover, we performed two experiments to get insights into the performance of some key components of the approach. The results are promising and indicate the ECo-IoT ability to enable the automated formation and adaptation of ECs at the scale of a few seconds.

In Paper IV, we proposed a commitment-based approach for enabling the automated formation and self-adaptation of ECs. It refines the abstract process presented in Paper III. More specifically, it exploits the concept of Commitments, adopted from the field of Multi-Agent Systems (MAS), and the concept of MAPE-K feedback loops, adopted from the field of Self-Adaptive Systems, to support autonomous things to negotiate agreements to realize ECs [61, 64]. We denoted ECs by the notation $EC = \langle A, C, S \rangle$ where A is a set of smart things, C is the set of commitments that hold among them and S is a set of state transitions that realize the user goal. Furthermore, we presented a conceptual model

for commitment-based ECs. We highlight that this approach can be integrated with both the IoT-FED and the ECo-IoT approaches presented above.

Compared to the related works discussed in Chapter 1.2.3, the processes proposed for realizing ECs provide IoT systems with more flexibility in responding to dynamic and uncertain environments.

4. Conclusions and Future Work

4.1. Conclusions

The type of applications enabled by the IoT and the rapid increase of the number of objects that get connected give to the IoT an increasingly important role in all aspects of our society. Although the IoT is well recognized in both academic and industrial communities, a widely accepted definition of the term is still elusive. The manifold visions about the IoT provide a fragmented picture, leading to a lack of common understanding about IoT systems. The involvement of human users and the dynamicity and uncertainty of the IoT environment put additional requirements on engineering IoT systems. In this thesis, we addressed the aforementioned challenges through three research questions. The first question concerns a characterization of IoT systems. The second and third research questions address suitable architectures and processes that enable users to achieve their goals by supporting the automated formation, enactment and adaptation of IoT systems in dynamic and uncertain environments. To answer these research questions, we proposed a number of contributions by applying two main research methods, i.e., the systematic mapping study and the design science research. More specifically, we systematically identified the characteristics of IoT systems by analyzing and synthesizing existing taxonomies. Also, we proposed novel approaches that exploit the concept of Emergent Configurations (ECs) as a basis to engineer goal-driven IoT systems that can cope with dynamic and uncertain environments.

4.2. Future Work

The contributions of this thesis can be extended in several directions addressing the different interesting areas as described below.

4.2.1. Characteristics of IoT systems

Based on the results presented in Paper I, more work is needed to investigate important characteristics of IoT systems such as user, goal, and deployment. In addition, efforts are needed to explore and analyze in

depth the quality aspects of IoT systems such as scalability, privacy and security. In addition, the proposed processes for designing IoT systems can be refined and evaluated in realistic case studies.

4.2.2. The Architectural Approaches

The proposed architectural approaches can be extended in several interesting directions including: proposing (intelligent) processes and techniques for effective and efficient user goal interpretation; and devising mechanisms to handle situations where concurrent ECs compete on available resources or aim to achieve contradicting goals. Additionally, more focus can be dedicated to the uncertainty aspect. Several challenges related to the provision of assurances for Self-Adaptive Systems in uncertain environments are presented in [65]. The approach proposed in Paper IV can be further developed to enable the realization of ECs in decentralized settings or to be integrated with the approaches proposed in Paper V and VI. We plan to contribute to the research directions described above, extend the prototypes developed in Papers V and VI, and perform more extensive evaluations.

4.2.3. User Involvement

To realize the ECs vision, novel interaction mechanisms are needed to enable users to express their goals, interact with the dynamically formed IoT systems, and to provide their feedback. Different technologies support users to interact with the IoT environment such as software applications, augmented reality, and gestures. The diversity and the number of connected things in surrounding environments put additional requirements on engineering usable interfaces for IoT systems. Therefore, mechanisms are needed to visualize those things, their capabilities, and to enable users to interact with them. In this context, several challenges have been presented in the literature [66]. Furthermore, investigations can be dedicated to devise models of human users behaviours that support IoT systems in learning and predicting user goals and actions.

4.2.4. Security and Privacy

As objects are getting connected everywhere, the security of IoT systems and the privacy of the people become major concerns [67]. ECs constituents can belong to different owners and can also be mobile. Therefore, mechanisms are needed to ensure that users are aware of the data collected about them, how it is used, and by whom. Furthermore, due to the dynamicity of the IoT environment and the involvement of human users, responsive and effective mechanisms are needed to secure communications among users and connected things. Several challenges, requirements, and issues related the security and privacy of IoT systems have

been proposed [68–70].

4.2.5. Deployment

ECs constituents are heterogeneous, sometimes mobile, and possibly autonomous. Taking decisions about where to deploy the processing and storage capabilities of ECs can be a complex task. Investigations can be dedicated to conducting trade off analysis among existing IoT deployment models to support IoT engineers to take design decisions concerning the deployment of ECs in particular and IoT systems in general [71].

REFERENCES

1. K. Ashton *et al.*, “That ‘internet of things’ thing,” *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.
2. D. Giusto, A. Iera, G. Morabito, and L. Atzori, *The Internet of Things: 20th Tyrrhenian Workshop on Digital Communications*. Springer Science & Business Media, 2010.
3. L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
4. E. Borgia, “The Internet of Things vision: Key features, applications and open issues,” *Computer Communications*, vol. 54, pp. 1–31, 2014.
5. D. Miorandi and S. Sicari and F. De Pellegrini and I. Chlamtac, “Internet of things: Vision, applications and research challenges,” *Ad Hoc Netw.*, vol. 10, no. 7, pp. 1497–1516, 2012.
6. H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, “The industrial internet of things (IIoT): An analysis framework,” *Computers in Industry*, vol. 101, pp. 1–12, 2018.
7. A. Brush, B. Lee, R. Mahajan, S. Agarwal, S. Saroiu, and C. Dixon, “Home Automation in the Wild: Challenges and Opportunities,” in *proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2115–2124, ACM, 2011.
8. V. Issarny, N. Georgantas, S. Hachem, A. Zarras, P. Vassiliadis, M. Autili, M. A. Gerosa, and A. B. Hamida, “Service-oriented middleware for the Future Internet: state of the art and research directions,” *Journal of Internet Services and Applications*, vol. 2, no. 1, pp. 23–45, 2011.

9. E. Silva, T. Batista, and F. Oquendo, "A Mission-Oriented Approach for Designing System-of Systems," in *System of Systems Engineering Conference (SoSE), 2015 10th*, pp. 346–351, IEEE, 2015.
10. F. Dalpiaz, A. K. Chopra, J. Mylopoulos, and P. Giorgini, "From Intentions to Social Commitments: Adaptation in Multiagent Systems," *Knowing, Reasoning, and Acting: Essays in Honour of Hector J. Levesque*, vol. 1, 2011.
11. A. Bucchiarone, M. D. Sanctis, A. Marconi, M. Pistore, and P. Traverso, "Design for Adaptation of Distributed Service-Based Systems," in *Service-Oriented Computing - 13th International Conference, ICSOC 2015*, pp. 383–393, 2015.
12. F. Alkhabbas, and R. Spalazzese, and P. Davidsson, "Architecting Emergent Configurations in the Internet of Things," in *Software Architecture (ICSA), 2017 IEEE International Conference on*, pp. 221–224, IEEE, 2017.
13. F. Ciccozzi and R. Spalazzese, "MDE4IoT: Supporting the Internet of Things with Model-Driven Engineering," in *International Symposium on Intelligent and Distributed Computing*, pp. 67–76, Springer, 2016.
14. D. Trcek, "Lightweight protocols and privacy for all-in-silicon objects," *Ad Hoc Networks*, vol. 11, no. 5, pp. 1619–1628, 2013.
15. A. Kees, A. M. Oberländer, M. Röglinger, and M. Rosemann, "Understanding the Internet of Things: A Conceptualisation of Business-to-Thing (B2T) Interactions," in *ECIS*, 2015.
16. B. Dorsemayne, J.-P. Gaulier, J.-P. Wary, N. Kheir, and P. Urien, "Internet of Things: a definition & taxonomy," in *Next Generation Mobile Applications, Services and Technologies, 2015 9th International Conference on*, pp. 72–77, IEEE, 2015.
17. J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.
18. T. Pflanzner and A. Kertész, "A Taxonomy and Survey of IoT Cloud

Applications,” *EAI Endorsed Transactions on Internet of Things*, vol. 3, no. 12, pp. Terjedelem–14, 2018.

19. L. M. Borges, F. J. Velez, and A. S. Lebres, “Survey on the Characterization and Classification of Wireless Sensor Network Applications,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1860–1890, 2014.
20. M. Bauer, N. Bui, J. De Loof, C. Magerkurth, A. Nettsträter, J. Stefa, and J. W. Walewski, “IoT Reference Model,” in *Enabling Things to Talk*, pp. 113–162, Springer, 2013.
21. P. Patel and D. Cassou, “Enabling high-level application development for the Internet of Things,” *Journal of Systems and Software*, vol. 103, pp. 62–84, 2015.
22. F. Ciccozzi, I. Crnkovic, D. Di Ruscio, I. Malavolta, P. Pelliccione, and R. Spalazzese, “Model-Driven Engineering for Mission-Critical IoT Systems,” *IEEE Software*, vol. 34, no. 1, pp. 46–53, 2017.
23. A. Gonzalez-Garcia, A. Alvarez-Alvarez, J. Pascual-Espada, O. Sanjuan-Martinez, J. M. C. Lovelle, and B. C. P. G-Bustelo, “Introduction to devices orchestration in Internet of Things using SBPMN,” *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 1, no. 4, 2011.
24. L. Barker, M. White, M. Curran, Z. Patoli, B. Huggins, T. Pascu, and N. Beloff, “Taxonomy for Internet of Things-Tools for Monitoring Personal Effects,” in *PECCS*, pp. 67–71, 2014.
25. Y. Mehmood, F. Ahmad, I. Yaqoob, A. Adnane, M. Imran, and S. Guizani, “Internet-of-Things-Based Smart Cities: Recent Advances and Challenges,” *IEEE Communications Magazine*, vol. 55, no. 9, pp. 16–24, 2017.
26. N. Srinidhi, S. D. Kumar, and R. Banu, “Internet of things for neophytes: A survey,” in *Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), 2017 International Conference on*, pp. 234–242, IEEE, 2017.
27. E. Siow, T. Tiropanis, and W. Hall, “Analytics for the Internet of Things: A Survey,” *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, p. 74, 2018.

28. S. Verma, Y. Kawamoto, Z. Fadlullah, H. Nishiyama, and N. Kato, "A Survey on Network Methodologies for Real-Time Analytics of Massive IoT Data and Open Research Issues," *IEEE Communications Surveys & Tutorials*, 2017.
29. M. Stolikj, "Building blocks for the Internet of Things," 2015.
30. L. F. Rahman, T. Ozcelebi, and J. J. Lukkien, "Choosing Your IoT Programming Framework: Architectural Aspects," in *Future Internet of Things and Cloud (FiCloud)*, 2016 IEEE 4th International Conference on, pp. 293–300, IEEE, 2016.
31. R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed Internet of Things," *Computer Networks*, vol. 57, no. 10, pp. 2266–2279, 2013.
32. D. E. Perry and A. L. Wolf, "Foundations for the Study of Software Architecture," *ACM SIGSOFT Software engineering notes*, vol. 17, no. 4, pp. 40–52, 1992.
33. D. Garlan and M. Shaw, "An Introduction to Software Architecture," in *Advances in software engineering and knowledge engineering*, pp. 1–39, World Scientific, 1993.
34. N. Medvidovic, and R. N. Taylor, "Software architecture: foundations, theory, and practice," in *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 2*, pp. 471–472, ACM, 2010.
35. R. N. Taylor, N. Medvidovic, and E. M. Dashofy, *Software Architecture - Foundations, Theory, and Practice*. Wiley, 2010.
36. L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*. Addison-Wesley Professional, 2003.
37. P. B. Kruchten, "The 4+1 View Model of Architecture," *IEEE software*, vol. 12, no. 6, pp. 42–50, 1995.
38. M. Bauer, M. Boussard, N. Bui, J. De Loof, C. Magerkurth, S. Meissner, A. Nettsträter, J. Stefa, M. Thoma, and J. W. Walewski, "IoT Reference Architecture," in *Enabling Things to Talk*, pp. 163–211, Springer, 2013.

39. H. Koziolok, A. Burger, and J. Doppelhamer, "Self-Commissioning Industrial IoT-Systems in Process Automation: A Reference Architecture," in *2018 IEEE International Conference on Software Architecture (ICSA)*, pp. 196–19609, IEEE, 2018.
40. R. Khan, S. U. Khan, R. Zaheer, and S. Khan, "Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges," in *Frontiers of Information Technology (FIT), 2012 10th International Conference on*, pp. 257–260, IEEE, 2012.
41. J. Kramer and J. Magee, "Self-Managed Systems: an Architectural Challenge," in *2007 Future of Software Engineering*, pp. 259–268, IEEE Computer Society, 2007.
42. J. P. Sousa and D. Garlan, "Aura: An Architectural Framework for User Mobility in Ubiquitous Computing Environments," in *Software Architecture*, pp. 29–43, Springer, 2002.
43. P. Spiess and S. Karnouskos and D. Guinard and D. Savio and O. Baecker and L.M.S. De Souza and V. Trifa, "SOA-based Integration of the Internet of Things in Enterprise Services," in *Web Services, 2009. ICWS 2009. IEEE International Conference on*, pp. 968–975, IEEE, 2009.
44. K. Dar, and A. Taherkordi, and R. Rouvoy, and F. Eliassen, "Adaptable Service Composition for Very-Large-Scale Internet of Things Systems," in *Proceedings of the 8th Middleware Doctoral Symposium*, p. 2, ACM, 2011.
45. R. De Lemos, H. Giese, H. A. Müller, M. Shaw, J. Andersson, M. Litoiu, B. Schmerl, G. Tamura, N. M. Villegas, T. Vogel, *et al.*, "Software Engineering for Self-Adaptive Systems: A Second Research Roadmap," in *Software Engineering for Self-Adaptive Systems II*, pp. 1–32, Springer, 2013.
46. H. Muccini, M. Sharaf, and D. Weyns, "Self-Adaptation for Cyber-Physical Systems: A Systematic Literature Review," in *Proceedings of the 11th international symposium on software engineering for adaptive and self-managing systems*, pp. 75–81, ACM, 2016.
47. Y. Brun, G. D. M. Serugendo, C. Gacek, H. Giese, H. Kienle, M. Litoiu, H. Müller, M. Pezzè, and M. Shaw, "Engineering Self-Adaptive Systems through Feedback Loops," in *Software engineering for self-adaptive systems*, pp. 48–70, Springer, 2009.

48. M. Hussein and S. Li and A. Radermacher, "Model-driven Development of Adaptive IoT Systems," in *4th International Workshop on Interplay of Model-Driven and Component-Based Software Engineering (ModComp) 2017 Workshop Pre-proceedings*, p. 20, 2017.
49. A. Ciortea, and O. Boissier, and A. Zimmermann, and A. M. Florea, "Responsive Decentralized Composition of Service Mashups for the Internet of Things," in *Proceedings of the 6th International Conference on the Internet of Things*, pp. 53–61, ACM, 2016.
50. R. Seiger, S. Huber, P. Heisig, and U. Aßmann, "Toward a framework for Self-adaptive Workflows in Cyber-Physical Systems," *Software & Systems Modeling*, pp. 1–18, 2017.
51. S. Hachem and A. Pathak and V. Issarny, "Service-Oriented Middleware for the Mobile Internet of Things: A Scalable Solution," in *IEEE GLOBECOM: Global Communications Conference*, 2014.
52. S. Mayer, and R. Verborgh and M. Kovatsch and F. Mattern, "Smart Configuration of Smart Environments," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 3, pp. 1247–1255, 2016.
53. J. W. Creswell, *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*, vol. 4. Sage Thousand Oaks, CA, 2009.
54. K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic Mapping Studies in Software Engineering," in *EASE*, vol. 8, pp. 68–77, 2008.
55. P. Johannesson and E. Perjons, *An Introduction to Design Science*. Springer, 2014.
56. R. C. Nickerson, U. Varshney, and J. Muntermann, "A method for taxonomy development and its application in information systems," *European Journal of Information Systems*, vol. 22, no. 3, pp. 336–359, 2013.
57. M. W. Maier, "Architecting Principles for Systems-of-Systems," in *INCOSE International Symposium*, vol. 6, pp. 565–573, Wiley Online Library, 1996.

58. J. Boardman and B. Sauser, "System of Systems-the meaning of of," in *System of Systems Engineering, International Conference on*, pp. 6–pp, IEEE, 2006.
59. C. B. Nielsen, P. G. Larsen, J. Fitzgerald, J. Woodcock, and J. Pelseska, "Systems of Systems Engineering: Basic Concepts, Model-based Techniques, and Research Directions," *ACM Computing Surveys*, vol. 48, no. 2, p. 18, 2015.
60. H.-D. Ma, "Internet of Things: Objectives and Scientific Challenges," *Journal of Computer science and Technology*, vol. 26, no. 6, pp. 919–924, 2011.
61. J. O. Kephart and D. M. Chess, "The Vision of Autonomic Computing," *Computer*, no. 1, pp. 41–50, 2003.
62. M. Ghallab, and D. Nau, and P. Traverso, *Automated Planning: theory and practice*. Elsevier, 2004.
63. C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context Aware Computing for The Internet of Things: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414–454, 2014.
64. M. P. Singh, *Social and Psychological Commitments in Multiagent Systems*. Citeseer, 1991.
65. J. Andersson, M. Litoiu, B. Schmerl, and D. Weyns, "Software Engineering for Self-Adaptive Systems: Research Challenges in the Provision of Assurances," in *Software Engineering for Self-Adaptive Systems III. Assurances: International Seminar, Dagstuhl Castle, Germany, December 15-19, 2013, Revised Selected and Invited Papers*, vol. 9640, p. 3, Springer, 2018.
66. M. Kranz, P. Holleis, and A. Schmidt, "Embedded Interaction: Interacting with the Internet of Things," *IEEE internet computing*, vol. 14, no. 2, pp. 46–53, 2010.
67. J. Bugeja, A. Jacobsson, and P. Davidsson, "On Privacy and Security Challenges in Smart Connected Homes," in *Intelligence and Security Informatics Conference (EISIC), 2016 European*, pp. 172–175, IEEE, 2016.

68. R. H. Weber, "Internet of Things—New security and privacy challenges," *Computer law & security review*, vol. 26, no. 1, pp. 23–30, 2010.
69. B. Fabian and O. Gunther, "Distributed ONS and its Impact on Privacy," in *Communications, 2007. ICC'07. IEEE International Conference on*, pp. 1223–1228, IEEE, 2007.
70. A. ur Rehman, S. ur Rehman, I. U. Khan, M. Moiz, and S. Hasan, "Security and Privacy Issues in IoT," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 8, no. 3, 2016.
71. M. Ashouri, P. Davidsson, and R. Spalazzese, "Cloud, Edge, or Both? Towards Decision Support for Designing IoT Applications," in *The Fifth International Conference on Internet of Things: Systems, Management and Security (IoTSMS 2018)*, 2018.

ISBN 978-91-7104-958-2 (print)

ISBN 978-91-7104-959-9 (pdf)

MALMÖ UNIVERSITY
205 06 MALMÖ, SWEDEN
WWW.MAU.SE