Chapter 1
Fog Computing Fundamentals in The Internet-of-Things

Behailu Negash, Amir M. Rahmani, Pasi Liljeberg, and Axel Jantsch

Abstract The functional separation of different system components has been used to address some critical challenges in architecture design. One of the well known approaches of physical separation of functional units is in client-server architectures. The server side of this separation is hidden inside the cloud infrastructure in the case of an Internet scale system. This model is serving a wide range of applications running over the Internet providing storage, computing power, and redundant services for reliability. However, in the new paradigm of the Internet-of-Things, the traditional separation partly fails to meet the set of system requirements. Fog computing is introduced as an intermediate layer between the clients and the Cloud. It brings computing, storage, management, and network services among others, closer to the sensor/actuator nodes. This book discusses the features of Fog computing, its advantages, internal details and present case studies to demonstrate it in real application scenarios. The focus of this chapter is to give an overview of Fog computing at a higher level.

1 Introduction

The Internet-of-Things (IoT) is a self-configuring and adaptive network which connects real-world things to the Internet enabling them to communicate with other connected objects leading to the realization of a new range of ubiquitous services [1]. This definition of IoT is not comprehensive. There is a variety of definitions that differ in details as reviewed in [1]. The term *IoT* originates to Massachusetts Insti-

*Behailu Negash, Pasi Liljeberg
University of Turku, Turku, Finland, e-mail: {behneg, pakrli@utu.fi}

Amir M. Rahmani
University of California, Irvine, USA and TU Wien, Vienna, Austria, e-mail: amirr1@uci.edu

Axel Jantsch
TU Wien, Vienna, Austria, e-mail: axel.jantsch@tuwien.ac.at

tute of Technology Auto-ID center when it was chosen by Kevin Ashton in 1999 [2]. However, the concept of connecting devices to the Internet to remotely monitor their status has been introduced for the first time in 1982 by a group of students at Carnegie Mellon University when they managed to connect a coke machine to the Internet and remotely check its status [3]. Advancements in science and technology enabled making smaller, cheaper and faster computing devices capable of sensing the environment, communicating and actuating remotely, which resulted to the increased interest of applying the IoT to vast aspects of life, such as smart cities, healthcare, and smart home. Some of these applications are discussed in details in the last part of this book in the case study section focusing on the benefits of Fog computing in the respective domains.

The IoT is already around us connecting wearable devices, smart cars and smart home systems. It is expected that more than 50 billion devices will be connected to the Internet by 2020 [4]. The introduction of such a huge number of connected devices requires a scalable architecture to accommodate them without any degradation of the quality of service demanded by applications. In addition, the majority of the devices that make up the Internet of Things are resource-constrained; resources, such as computing power, energy, bandwidth and storage are scarce. These constraints limit the deployment scenarios of applications using such IoT devices. For instance, it is infeasible to use a battery-powered sensor to directly connect to the Internet and publish information regarding its surrounding for a long time or store readings of a longer time in local memory. These constraints present a design challenge that is shaping the architecture of the IoT in many ways. Some of these IoT challenges and coresponding efforts in each area are summarized in [5]. Many of these challenges can be mitigated by extending the functions of Cloud computing closer to the IoT devices. Fog computing [6], also known as edge computing, is such an intermediate layer extending the Cloud layer.

Fog computing layer brings computing, network and storage services closer to the end-nodes in IoT. Compared to Cloud computing, this computing layer is highly distributed and introduces additional services to end-devices located in the perception layer [7]. This bridging layer is referred differently but with similar or small variations in purpose. For example, edge computing [8, 9], micro-cloud [10] or cloudlet[11] are some of the related terms. Regardless of the name, the concept of introducing an intermediate computing layer in IoT is motivated by the similar set of challenges. Moreover, the possible set of services that can be potentially integrated in the Fog computing layer are vast. Some of these services are a scaled version of the ones provided by cloud computing while most of these services emerged recently in response to IoT challenges. This chapter goes through the basics of the requirements of IoT which motivate the benefits of Fog computing, and discusses some features, organization, and functions of this layer. Subsequent chapters explain these concepts in more details and give specific implementation scenarios and use cases of Fog computing. To show the overall organization of the book, we briefly discuss the contribution of each chapter in the end of this introductory chapter.

2 Background and Motivation of Fog Computing

The architecture of the Internet-o-Things is an active research area. Architecture plays a critical role in determining the success of a system. As such, there are several efforts ranging from public projects to industrial standard associations and academic institutions to set a working IoT architecture. Some of these efforts are presented in [12]. Most of the architecture proposals are generic and model IoT regardless of the specific application domain or implementation. The most notable project, Internet of Things Architecture (IoT-A) [13] provides a generalization of IoT domain model that serves as basis for a reference architecture. In the functional view of the model, IoT-A identifies components of an IoT system; for instance, communication, security, management, and IoT services as the main elements. Another generalized component view of IoT systems is presented in [5] where Al-Fuqaha *et al.* discuss IoT as a integration of identification, sensing, communication, computation, services and semantics. These modular classifications of IoT is based on the functionality of each unit. Some of these units can be located in a single device. However, an IoT system is naturally a distributed system by definition. Hence, the components identified above are geographically distributed where the communication component is in charge of connecting them. In the simplest form, two groups can be formed: the first group contains identification and sensing while the second one hosts computation, services and semantics (Similar separation can be also achieved in case of IoT-A model). In an effort to find the best level of functional categorization and physical separation, researchers have proposed different alternatives.

A straight forward way to make an IoT device visible through the Internet is to provide it with an access to a Cloud server, such that it can upload data, receive notifications or commands. In such configuration, the client handles reading data from the environment and most of the remaining functions run in the Cloud. This traditional client-server approach of organizing the different components of IoT is still used by many vendors. There are also many variations of this architecture to separate certain logical components of the system into three or five layers [5] (Figure 1). These separations of concerns are mostly based on the functionality of the module. In the three layer architecture, the sensors appear in the lower perception layer. Network layer, which is located on top of perception layer, connects the sensors to the top most Application layer. The functionality of each layer is distinct in this approach. The sensors and actuators in the perception layer gather the data that will be transported through the network to ultimately reach the application logic. Figure 1 show the different types of logical separation of IoT elements. Other alternatives of this architecture proposal divide the layers into five. Some of this variations consider middleware and object abstraction as separate layers. These additional layers help provide integration services and encapsulate the devices in the perception layer respectively. Even though, implementing these layers of logically separated components as such provides modularity and ease of implementation, it fails to address the requirements of the perception layer, such as low latency communication and mobility.

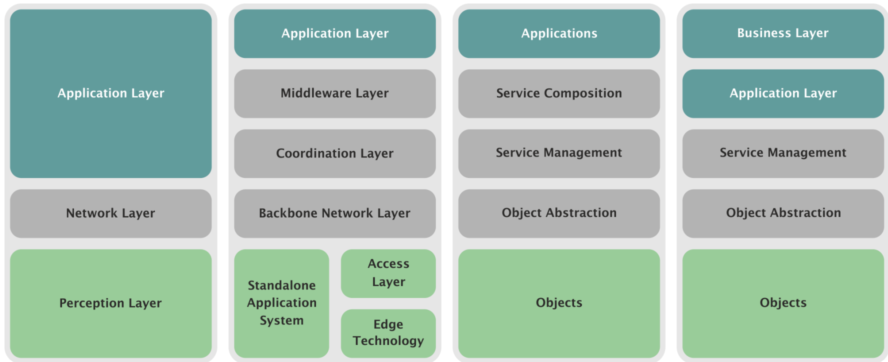

Fig. 1 IoT architecture proposals (3-layers and 5-layers) [5]

The perception layer or Sensor layer, shown in Figure 2, can be composed of millions of devices. The majority of these devices are very tiny in size, battery-powered, have small memory and limited processing power. Such resource constraints necessitate novel design approaches to accommodate them. In addition, various wireless communication protocols are widely used for networking such as Wi-Fi, Bluetooth Low Energy, NFC, Zigbee, RFID, 6LoWPAN. Besides the foregoing network protocol variations, there are differences in application layer protocols even among devices using same underlying network protocol. For instance, CoAP [14], MQTT [15], DDS [16] and XMPP [17], are among the frequently used ones. Furthermore, there are multiple data formats used by these protocols that are application domain specific. The resource constraints mentioned above, the heterogeneity of protocols, platforms, and data formats calls for the design of more efficient and IoT-friendly architectures.

The design process of a concrete architecture for a system depends on the attributes of the specific application. However, based on the generic IoT challenges and requirements highlighted above, it is possible to have a reasonable generic architecture design. Following on the logical separation of functional components that resulted in three or five layers, we can map the logical components to physical computing layers. As mentioned earlier, in a client server approach, most of the components (shown in Figure 1) would run in the server located in a cloud. Unfortunately, this approach does not address all the requirements discussed above. This initiated the research for an alternative computing hierarchy that works well for IoT. Fog computing is introduced as an intermediate layer between the perception layer and the Cloud, giving more flexibility of choice for deploying the components of an IoT system architecture. Figure 2 shows how Fog computing fits between the perception layer or sensors and the Cloud layer. In the following sections, this layer is discussed by giving more details on the internal organization and services provided.

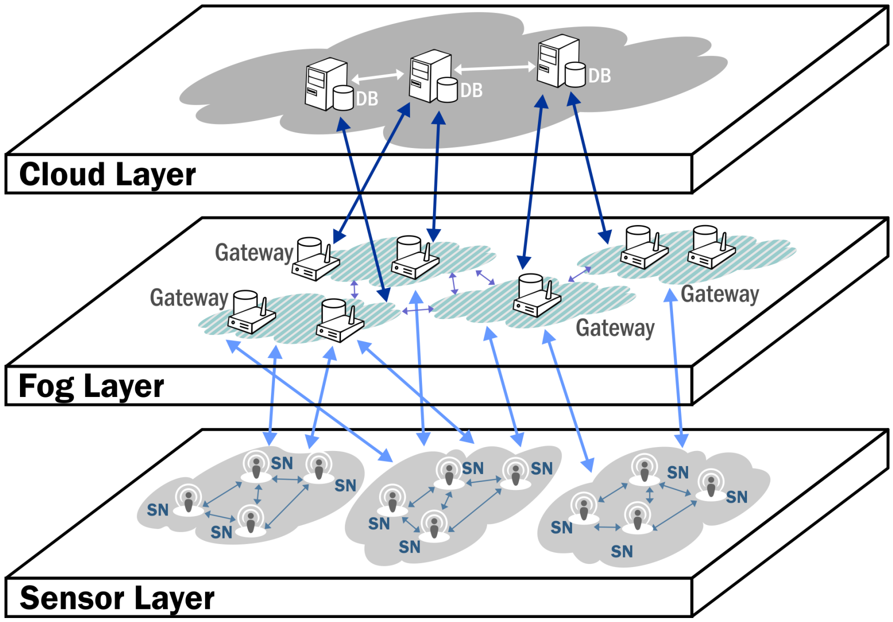

Fig. 2 High level overview of fog based IoT

3 Fog Computing Basics

The introduction of the IoT brings billions of devices to the Internet and the majority of these devices are resource-constrained. To overcome the challenges of these devices and meet the requirements of the application domain, the demand for an intermediate computing layer becomes evident. The concept of fog computing is the latest descendant in the line of physical separation of functional units. It is a computing layer closer to the perception layer, where the sensors and actuators reside, and provides computing, networking, and storage services. To accommodate these services, and address the requirements of IoT systems, the Fog layer offers the Characteristics discussed in the following.

3.1 Characteristics of the Fog layer

As an intermediate computing layer, the characteristics of the Fog layer are discussed in comparison to the perception and Cloud layers. In contrast to the Cloud layer, the Fog layer is closer to the perception layer and this proximity provides a range of advantages that characterise the layer. One of the immediate benefits over the Cloud is its location-awareness. Such awareness comes due to the large scale geographical distribution of the devices that make up the Fog layer [6]. As shown in Figure 2, each gateway in the Fog layer manages a subset of nodes in the perception or sensor layer. This subset of resource constrained devices are located close to each other and the managing gateway can easily locate each device. The location-awareness of the Fog layer can be utilized to address multiple functional and non-functional requirements of IoT applications, such as mobility and security. Another closely related characteristics of the Fog layer is its large scale distribution in contrast to the centralized Cloud layer. Centralization in this context is relative; the Cloud layer is centralized as seen from the client side. Looking from the organization of the servers in the cloud, however, it is geographically distributed but not at the scale expected from the Fog layer. For instance, Cloud service providers such as Amazon have multiple data centers in different regions. The case of geographical distribution in the Fog layer is different due to the small separation distance of the gateways and its large deployment.

The combined benefits of location awareness and large scale geographic distribution support the mobility requirements of devices or “things” at the perception layer. An overview of the services available to provide mobility is discussed in the following sections and its practical applications are presented in the later chapters. Moreover, the close-proximity of the Fog layer to the nodes provides real-time interaction mode with the sensors and actuators in the perception layer. The geographical distribution of the Fog layer and subsequently the offered low communication latency are among the critical features of fog layer. Some IoT application domains, such as healthcare or automotive, are highly dependent on such feature. For instance, a case study of edge-based ECG feature extraction is presented in [18].

The IoT in general is dominated by wireless networks. There are many wireless protocols, mostly tailored for low-power operation, coverage or bandwidth. For instance, 6LoWPAN [19], Bluetooth Low Energy (BLE), NarrowBand IoT Protocol (NB-IoT) [20], LoRa [21] and Sigfox [22] are some of these protocols. The majority of these protocols connect sensor nodes to the Fog layer to get access to the Internet. These protocols are usually incompatible with each other. To cope with this issue, Fog layer provides an additional benefit of acting as an interoperability layer among these heterogeneous protocols. There are several middleware proposals that utilize this layer as a means to translate or adapt different network or application protocols [23]. The gateways in the Fog layer can also perform lightweight analytics at the edge to give feedback, command, and notification to the end-users as well as the sensor nodes in real-time. In addition, the internal organization of the Fog itself can be arranged in a federated or hierarchical way based on functional or location of the connected devices.

3.2 Design and Organization of the Fog layer

Based on the characteristics of the Fog layer presented in the previous section and the possible set of services highlighted in the following section, the Fog layer can be organized in an efficient way to address the requirements. This section is by no means comprehensive and detailed enough for build a usable intermediate layer, but instead gives an introductory information that will be dealt in more details in later chapters. To begin with, consider a network gateway or a wireless hot-spot serving clients in its vicinity. The role of such a gateway is to pass network packets to the back-end infrastructure which is connected to the Internet. In a larger environment, multiple access-points can be arranged to provide users with seamless connectivity throughout the intended area. Considering the tons of devices that connect to the Fog layer, this layer can be visualized as a network of gateways covering a larger area. In addition to simply passing network packets, these networked smart gateways can process the data or store it when necessary [18]. Figure 2 shows the Fog layer where distributed smart gateways communicate with the Cloud, the sensor layer, and among themselves. In the gateways of the Fog layer, the network interface is a critical component to enable support for the various wireless network protocols shown in Section 3.1.

4 Fog Computing Services

The characteristics of Fog computing layer has been highlighted in Section 3.1. We mentioned that these characteristics can be leveraged to provide services that assist the perception layer, such that the overall system requirements are met. This layer takes advantage of its proximity to the sensor layer and provides services that are extensions of the cloud layer and also unique ones that are feasible only at this layer. This section gives an overview of a subset of possible services at the fog layer and the related advantages that enable IoT. These services are organized into Compute, Storage and Network services.

4.1 Computing services

The limitations of computing power of the devices in the perception layer has led to the introduction of remote processing approaches. Processing at the Fog layer is not only motivated by the constraint of processing power at sensor nodes, but also by the desired location of computing to better meet system requirements and maintain energy efficiency. Earlier Cloud-based processing can be brought down to the Fog layer for localized processing and immediate response [24, 25]. In this regard, there can be multiple configurations of sharing the computing load among the different layers in the IoT-based system, and the processing requirements may vary based on the actual work. For instance, considering a system which performs data processing to learn a certain pattern, the workload can be distributed in such a way that localized patterns can be identified in the Fog layer while the generalized patterns is only available at the Cloud. This load sharing is discussed in detail in upcoming chapters.

Beside data management, events can be handled at the Fog layer. The proximity of this layer makes it an ideal candidate to handle events to react in real-time and enhance the reliability of the system. Moreover, there are many middleware that leverage the Fog layer to manage physical devices through abstraction, agent-based management, and virtual machines. The following sections provide an introductory overview of these computing services that can be realized at the Fog layer.

4.2 Storage Services

A huge amount of data can be generated by the sensor nodes and there are billions of these sensor devices around. The storage available in the devices at the perception layer is not often sufficient to store even a one-day data considering the rate of data generation. As discussed earlier, pushing all the data directly to the cloud is not necessary in particular when there is irrelevance or redundancy in data. The wise approach in such cases would be to filter and temporally store the data in the intermediate fog layer [26, 27]. Combined with the computing service, the stored data can be filtered, analyzed, and compressed for efficient transmission or for learning local information regarding the system behaviour. In cases where the communication may not be robust, the storage services help enhance the reliability of the system by maintaining proper system behavior for client nodes. Sarkar *et al.*[28] present such features of fog layer in their assessment of the fog computing for IoT.

4.3 Communication Services

The communication in the Internet-of-Things is dominated by wireless nodes. Due to the resource constraints in the perception layer, these wireless protocols are optimized for low power operation, narrow-band transmission or longer range of coverage. Currently, a long list of alternative protocols are available in the market [29]. The Fog layer is located in a strategic place to organize these multitude of wireless protocols and unify their communication to the Cloud layer. This helps in managing sub-networks of sensors and actuators providing security, channeling messages among devices and enhancing the reliability of the system. In addition, this layer can provide interoperability of disparate protocols by listing and interpreting the representation format. Moreover, the Fog layer provides visibility of devices that are non-IP-based to be accessible through the Internet [26].

5 Summary and the Book Organization

This chapter presented a concise introduction of Fog computing in Internet-of-Things. It covered the driving reasons for the design of IoT systems reinforced with the fog computing as an intermediate computing layer. In addition, a high level introduction of the internal structure and organization of the layer was discussed. As a motivation to the demand for Fog computing, we showed the functional units of an IoT system and distribution of these units through physical computing layers. In doing so, however, IoT system requirements such as mobility and resource constraints of the perception layer demand a nearby layer. This layer, introduced as Fog due to its proximity to the ground compared to the Cloud in its literal meaning, provides connectivity, storage and processing for sensors and actuators. To achieve these functions and scale, the Fog layer has modular organization and is geographically distributed. These provided services via Fog Computing are classified into three main classes as compute, storage and network functions supporting the sensor nodes.

The heterogeneous nature of the wireless network protocols, platforms and architectures in the perception layer of IoT makes it difficult to build an integrated and reliable system. The Fog layer provides services that can be used to hide such heterogeneity and provide a uniform access channel to the perception layer for users through the Internet. These concepts are well examined with practical implementations and evaluation of certain aspects of their performance in upcoming chapters.

This book is organized in eight chapters to provide readers with a comprehensive information about fog computing in the context of Internet-of-Things. Chapter 1 is a preliminary overview of what fog computing is, its characteristics, and the possible set of services that can be hosted in this layer to meet system requirements of IoT. In essence, it gives a general background that serves as foundation to understand subsequent chapters. Figure 3 shows the organization of the book in general. This chapter gives wider but shallower conceptual basis that are enforced with two chapters going into details of the internal structure of Fog computing focusing on management. Chapter 2 and 3 give details on Fog based IoT scalability and resource estimation of the Fog layer to accommodate the billions of additional Internet connected devices. The third part of the book contains three chapters that go further in discussing some of the critical services needed at the Fog layer. Chapter 4, 5, and 6 focus on security and privacy of the Internet of Things as realized through the Fog layer, learning and self-awareness of IoT systems through Fog computing, and detailed data analytics in a smart city application.

The last part of the book has two chapters that discuss specific application scenarios of IoT systems and the advantage of Fog computing in those domains. Electrical grid control system implementation and healthcare are two application areas explained with concrete application scenarios in Chapter 7 and 8, respectively.

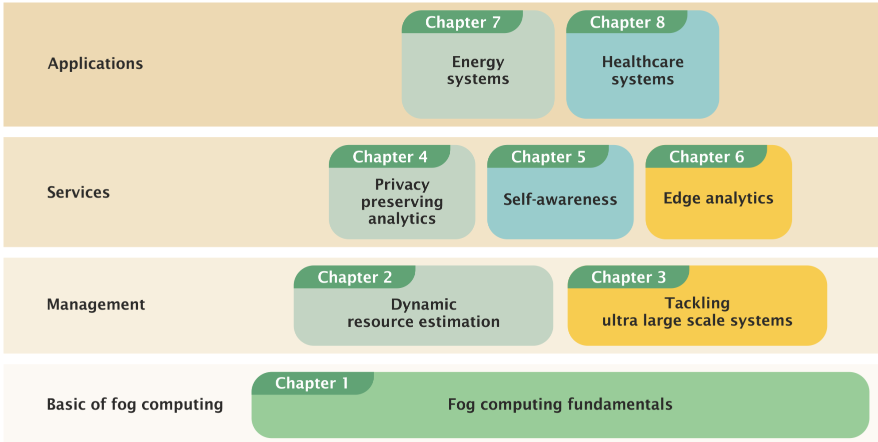

Fig. 3 Organization of the chapters

References

1. Roberto Minerva, Abyi Biru, and Domenico Rotondi. Towards a definition of the internet of things (IoT). Technical report, IEEE, 2015.
2. Kevin Ashton. That 'Internet of Things' Thing: In the real world, things matter more than ideas, RFID Journal. http://www.rfidjournal.com/articles/view?4986.
3. CMU. The "only" coke machine on the internet. http://www.cs.cmu.edu/ coke/.
4. S. Ray, Y. Jin, and A. Raychowdhury. The Changing Computing Paradigm With Internet of Things: A Tutorial Introduction. *IEEE Design Test*, 33(2):76–96, April 2016.
5. A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys Tutorials*, 17(4):2347–2376, Fourthquarter 2015.
6. Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, MCC '12, pages 13–16, New York, NY, USA, 2012. ACM.
7. Shanhe Yi, Cheng Li, and Qun Li. A survey of fog computing: Concepts, applications and issues. In *Proceedings of the 2015 Workshop on Mobile Big Data*, Mobidata '15, pages 37–42, New York, NY, USA, 2015. ACM.
8. F. Jalali, A. Vishwanath, J. de Hoog, and F. Suits. Interconnecting fog computing and micro-grids for greening iot. In *2016 IEEE Innovative Smart Grid Technologies - Asia (ISGT-Asia)*, pages 693–698, Nov 2016.
9. O. Salman, I. Elhajj, A. Kayssi, and A. Chehab. Edge computing enabling the internet of things. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 603–608, Dec 2015.
10. M. Selimi, L. Cerdà-Alabern, L. Wang, A. Sathiaseelan, L. Veiga, and F. Freitag. Bandwidth-aware service placement in community network micro-clouds. In *2016 IEEE 41st Conference on Local Computer Networks (LCN)*, pages 220–223, Nov 2016.
11. M. Satyanarayanan, P. Simoens, Y. Xiao, P. Pillai, Z. Chen, K. Ha, W. Hu, and B. Amos. Edge analytics in the internet of things. *IEEE Pervasive Computing*, 14(2):24–31, Apr 2015.
12. S. Krčo, B. Pokrić, and F. Carrez. Designing iot architecture(s): A european perspective. In *2014 IEEE World Forum on Internet of Things (WF-IoT)*, pages 79–84, March 2014.
13. IoT-A Project. Internet of things - architecture, IoT-A, deliverable d1.5 - final architecture reference model for the IoT v3.0. Technical report, EU-FP7, 2013.

14. Z. Shelby, K. Hartke, and C. Bormann. The constrained application protocol (CoAP). https://tools.ietf.org/html/rfc7252.
15. OASIS. MQTT version 3.1.1 oasis standard. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.pdf.
16. OMG. Data distribution service DDS. http://www.omg.org/spec/DDS/1.4/.
17. XSF. Extensible messaging and presence protocol (XMPP). https://xmpp.org/extensions/index.html.
18. T. N. Gia, M. Jiang, A. M. Rahmani, T. Westerlund, P. Liljeberg, and H. Tenhunen. Fog computing in healthcare internet of things: A case study on ecg feature extraction. In *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, pages 356–363, Oct 2015.
19. N. Kushalnagar, G. Montenegro, and C. Schumacher. Ipv6 over low-power wireless personal area networks (6LoWPANs): Overview, assumptions, problem statement, and goals. https://tools.ietf.org/html/rfc4919.
20. Y.-P. Eric Wang, Xingqin Lin, Ansuman Adhikary, Asbjörn Grövlen, Yutao Sui, Yufei W. Blankenship, Johan Bergman, and Hazhir Shokri-Razaghi. A primer on 3gpp narrowband internet of things (nb-iot). *CoRR*, abs/1606.04171, 2016.
21. LoRa Alliance. LoRa wide area network for iot. https://www.lora-alliance.org/What-Is-LoRa/Technology.
22. Sigfox. About sigfox. http://www.sigfox.com/.
23. Behailu Negash, Amir M. Rahmani, Tomi Westerlund, Pasi Liljeberg, and Hannu Tenhunen. LISA 2.0: lightweight internet of things service bus architecture using node centric networking. *Journal of Ambient Intelligence and Humanized Computing*, 7(3):305–319, 2016.
24. S. K. Datta, C. Bonnet, and J. Haerri. Fog computing architecture to enable consumer centric internet of things services. In *2015 International Symposium on Consumer Electronics (ISCE)*, pages 1–2, June 2015.
25. P. Hu, H. Ning, T. Qiu, Y. Zhang, and X. Luo. Fog computing-based face identification and resolution scheme in internet of things. *IEEE Transactions on Industrial Informatics*, PP(99):1–1, 2016.
26. A. M. Rahmani, N. K. Thanigaivelan, Tuan Nguyen Gia, J. Granados, B. Negash, P. Liljeberg, and H. Tenhunen. Smart e-health gateway: Bringing intelligence to internet-of-things based ubiquitous healthcare systems. In *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, pages 826–834, Jan 2015.
27. A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, and P. Liljeberg. Exploiting smart e-health gateways at the edge of healthcare internet-of-things: A fog computing approach. *Future Generation Computer Systems*, 2017.
28. S. Sarkar, S. Chatterjee, and S. Misra. Assessment of the suitability of fog computing in the context of internet of things. *IEEE Transactions on Cloud Computing*, PP(99):1–1, 2015.
29. Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. Mccann, and K. K. Leung. A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities. *IEEE Wireless Communications*, 20(6):91–98, December 2013.