UNIVERSITA’ DEGLI STUDI DI CATANIA

DOTTORATO IN INGEGNERIA DEI SISTEMI, ENERGETICA, INFORMATICA E DELLE TELECOMUNICAZIONI

Edge Computing: challenges, solutions and architectures arising from the integration of Cloud Computing with Internet of Things

Ermanno Guardo

Coordinatore
Prof. Paolo Arena

Tutor
Prof. Aurelio La Corte

Contents

List of Figures

Abstract

The rapid spread of the Internet of Things (IoT) is causing the exponential growth of objects connected to the network, in fact, according to estimates, in 2020 there will be about 3/4 devices per person totaling of over 20 billion connected devices. Therefore, the use of content that requires intensive bandwidth consumption is growing.

In order to meet these growing needs, the computing power and storage space are transferred to the network edge to reduce the network latency and increase the bandwidth availability.

Edge computing allows to approach high-bandwidth content and sensitive apps to the user or data source and is preferred to use it for many IoT applications respect to cloud computing. Its distributed approach addresses the needs of IoT and industrial IoT, as well as the immense amount of data generated by smart sensors and IoT devices, which would be costly and time-consuming to send to the cloud for processing and analysis. Edge computing reduces both the bandwidth needed and the communication among sensors and cloud, which can negatively affect the IoT performance.

The goal of edge computing is to improve efficiency and reduce the amount of data transported to the cloud for processing, analysis and storage.

The research activity carried out during the three years of the Ph.D. program focused on the study, design and development of architectures and prototypes based on the Edge Computing in various contexts such as smart cities and agriculture. Therefore, the well-known paradigms of Fog Computing and Mobile Edge Computing have been faced.

In this thesis, will be discussed the work carried out through the exploitation of the Fog Computing and Mobile Edge Computing paradigms, considered suitable solutions to address the challenges of the fourth industrial revolution.

1 Introduction

The concept of Cloud Computing has been evolving over the years since its introduction. Although it was introduced from the mainframe model, the Cloud Computing concept expanded from ’60 and ’70 including not only a processors sharing, but also other concepts and technologies.

In 1961, John McCarthy at MIT’s centennial celebration stated that “Computing may someday be organized as a public utility just as the telephone system is a public utility”, thus imagining a future where Computing could have been distributed and organized on different systems of public access [1].

Cloud Computing regards both applications delivered as services over the Internet, and the hardware and software systems in the data center that provide those services [2].

The benefits of Cloud Computing have been discussed since a long time, but now we are witnessing the fourth industrial revolution related to the Internet of Things (IoT), the era in which “things” tend to gain more and more intelligence, becoming smart and being able to communicate with other “things”, integrating several technologies and communications solutions [3].

IoT is a remarkable transformation of the way in which our world is interacting [4]. Much like the World Wide Web connected computers to networks, and the next evolution connected people to the Internet and to other people, the IoT can interconnect devices, people, environments, virtual objects, machines and internet services with the goals of reducing the complexity of creating opportunities for a closer integration of the physical world with computer-based systems and developing efficient solutions for smart city services. Developing smart city services [5] such as transportation, parking, lighting, traffic, waste and safety requires the availability of a platform which permits speeding up and reducing the implementation costs of the services themselves, providing new capabilities for automation, analysis at multiple levels, greater scalability and virtualization.

Therefore, the IoT paradigm is the key towards this revolution improving service levels and as a consequence the customer satisfaction, and meeting the demand of a new generation of empowered customers with smart products. IoT modeling constitutes a global network of interconnected and uniquely addressable "things", merging various heterogeneous communication technologies, both wired and wireless [6].

IoT will not be seen as individual system, but as a critical, integrated infrastructure upon which many applications and services can run [7].

The extended use of heterogeneous sensors, involves a new challenge to extract useful information from a complex sensing environment [8]. Internet of Things is leading the research to investigate and develop novel and high-performance computing architectures due to large amount of data analysis produced. Cloud Computing provides a solution to support dynamic scalability in many vertical areas such as smart city [9], [10], [11], smart home [12], smart agriculture [13], healthcare [14] [15], smart grid and several context-aware environment of Wireless Sensor Networks (WSNs) [16].

However, the deployment of a large number of devices and sensors for IoT requires location awareness and low latency, which, currently, are missing in commercial Cloud Computing models. One of the main challenges of IoT is related to increasing amount of devices connected to the network which is going to further grow [17], [18].

Therefore, Cloud Computing must be able in the near future to manage this large amount of data from the network that will probably lead to the reduction of available bandwidth and the subsequent increase of latency. IoT, along with actual and future challenges of Cloud Computing, claims for novel frameworks and paradigms, able to face the ubiquitous and pervasive nature of networks and the data-intensive Computing requirements [19].

To overcome this hurdle, has been conceived the edge computing paradigms such as Fog Computing [20] and Mobile Edge Computing (MEC) [22].

Strictly linked to IoT, the Fog Computing broadens the Cloud, extending some services also to the edge of network with the aim to transfer a part of processing at the edge of the network, close to the users and the field devices.

Thus, Fog Computing founded as a highly virtualized platform and applicable in many IoT scenarios, moves the processing abilities closer to the data source, proving useful to the smart grid, realizing more rapid M2M (Machine-to-Machine) communications, and even in the smart city context, shifting the decisions near the place where data are collected [20]. Therefore, in order to avoid a high exploitation of the Cloud resources, Fog Computing applications and services, allow interaction between Cloud and Fog, in particular when it comes to data management and analytics [21].

Whereas the Fog Computing is focused about edge computing concept with a more general point of view and can be designed on different layers of the network architecture, the MEC instead is more specific and consists of assigning computing capability to the RAN (Radio Access Network) in such a way to make a pre-processing of data from users/things which are within the range of the specific BTS (Base Transceiver Station) providing rapid deployment of applications and other customer services. The main idea behind Mobile Edge Computing is to reduce network congestion and improve applications by performing related processing tasks closer to the end user.

MEC can be implemented both indoors and outside depending on the access technology. With respect to the outdoors, macro cells place computing and virtualization capabilities into radio network elements.

In this thesis, various edge computing solutions (architectures and applications) will be discussed covering different contexts such as smart city, agriculture, and critical event management.

The proposed solutions concern both Fog Computing and Mobile Edge Computing paradigms.

The thesis is organized as follows: at the beginning it is explained the fourth industrial revolution highlighting the impact of the IoT in the main scenarios in which it is spreading and the protocols used. Chapter 3 discusses the edge computing paradigm focused on fog computing and mobile edge computing.

Chapter 4 discusses on the work carried out. Finally, Chapter 5 concludes with some considerations and discussions of future work.

2 The fourth industrial revolution

2.1 Impact of IoT

The IoT represents the new evolutionary stage of the Internet: no more people, but things, connect and interact through the network, improving the ability to collect, analyze and distribute convertible data into information. So new scenarios are opened for businesses and for people.

IoT already has impact on our daily lives. It will continue to generate significant and broad ranging impact across all sectors making life easier and smarter which confirms the view that the IoT has a wide variety of meanings, and that its impact will vary across different companies and industry sectors.

Figure 1- Infographic which represent the fourth industrial revolution

2.2 IoT Segments

2.2.1 Smart City

Smart city spans through a wide range of use cases, e.g. traffic management, water distribution, waste management, urban security, environmental monitoring, etc. Its popularity is fueled by the fact that many Smart city solutions promise to alleviate real pains of people living in cities these days and improve QoL (Quality of Life). IoT solutions in the area of Smart city solve traffic congestion problems,

reduce noise and pollution and help make cities safer. According to World Health Organization (WHO), "The global urban population is expected to grow by approximately 1.84 percent per year between 2015 and 2020 and by 1.63 percent per year between 2020 and 2025" [23]. Considering these alarming figures - both in terms of urban overpopulation and rural under population, there is a need to build smarter cities that can accommodate traffic smartly by reducing traffic congestion and air pollution efficiently. IoT will enable smooth flow of traffic in busy cities, allowing you to spend quality time with your family, rather than on the road and most importantly, decreasing the number of fatalities every year. This could be achieved by leveraging analyzed data for real-time solution and advanced communication systems using next generation technologies.

2.2.2 Smart Agriculture

Agricultural IoT will completely subvert the traditional assertions that "physical world" and "ICT world" are separated. In agricultural IoT, farmland, agricultural machineries, and fresh agricultural products are integrated with the chips, broadband network and database systems, forming a completely new “agricultural infrastructure”. IOT applications in agriculture include food traceability (RFID), soil and plant monitoring, precision agriculture, greenhouse environment monitoring and control systems, monitoring of food supply chain, monitoring of animals, etc.

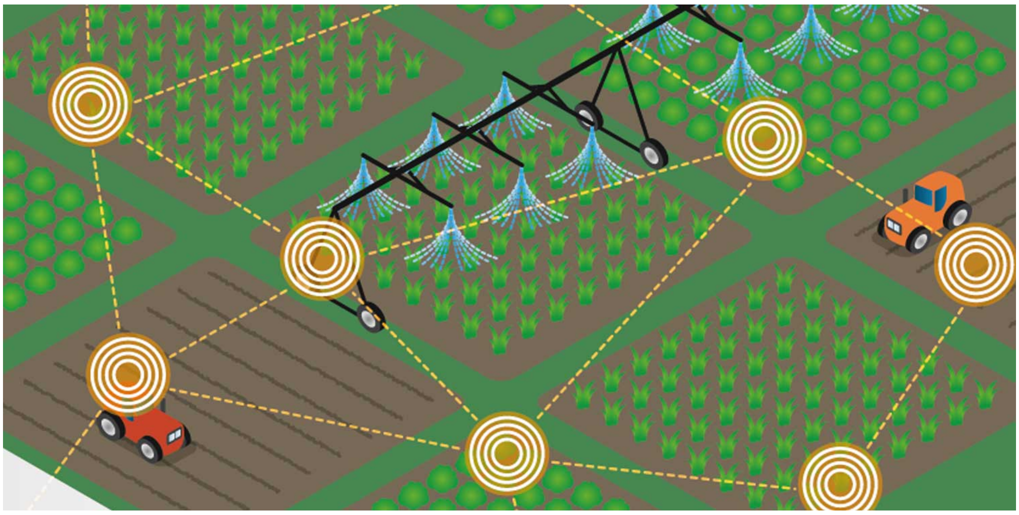

Figure 2 – Example of smart agriculture scenario

2.2.3 Smart Home

Smart homes are the next big thing driven by IoT. Imagine your home equipped with smart things, like home appliances, refrigerators and washing machines that can send alerts based on the situation and requirement. Air conditioners and heaters are synchronized with electronic sensors to minimize consumption of energy, providing environmental benefits. Also, smart homes will be built for elderly care to monitor their health and call for help, as and when required. Thus, giving the elderly chance to stay longer in their own home and environment.

Figure 3 - Major services in a smart home scenario

2.2.4 Smart Healthcare

Internet - connected devices are steadily being introduced in the healthcare sector in various forms and segments. Using the IoT concept to the area will make it possible to realize instant and continuous monitoring, monitoring and remote adjustments and treatment, etc. in cases such as breathing patterns, temperature, blood pressure, physical position and balance. These innovative IoT - based wearable devices are helping users minimize healthcare challenges, thus offering a better quality of life (QoL). For example, AliveCor [24] wireless heart monitor allows patients to monitor and manage their cardiac conditions. These services will create continuous revenue through consumption or subscription fees for continuous data analysis. Also, creating an opportunity for smarter devices to deliver valuable data, while reducing the need for direct patient - doctor interaction. The impact of IoT on healthcare will be very high. It will revitalize better and faster treatment, shorter and better recovery, improved QoL, and longer stay in own home, safety, cost savings, maintaining healthcare plans , keeping citizens fit , more efficient treatment, less hospital beds and places in elder centers, less hospitalization and cost savings in reduced consultations and shorter use of recovery equipment.

2.2.5 Smart Grid

To realize the full potential of Smart Energy, major changes in the energy system, specially at the lower network level, will require integration of smart houses in the smart grid. IoT will play a vital role in streamlining the transfer of high volumes of data over IP. IoT connected devices will establish communication between context - aware sensors and smart meters at user sites, allowing devices to be switched off based on load patterns. Smart grid technology is helping businesses and consumers cut their energy usage while advancing an intelligent, resilient and self - balancing utility network by enabling utilities to wirelessly connect to circuit breakers and meters. Real - time data from smart meters is also helping utilities monitor use and tackle energy efficiency. A smart grid is mostly about balance and efficiency while optimally delivering energy at the lowest cost and highest quality. For instance, Pacific Gas and Electric installed smart meters that help to cut the duration of power outages, improve operations by seeing energy consumption on one dashboard. The last shows where energy demands will surge enabling them to redistribute energy accordingly. Customers can now track their energy usage using a website that is in - sync with the smart meter sitting outside their home.

Figure 4 - Example of Smart Grid Scenario

2.2.6 Smart Connected Car

IoT enabled cars are coming slowly and steadily due to the fact that the automotive industry takes 2 - 4 years. These cars of the future will be very different from the cars of today. Connected, automated vehicles will communicate among themselves and / or to Infrastructure through Vehicle - to - Vehicle (V2V) and Vehicle - to - Infrastructure (V2I) technology and will reform road safety and reduce accidents. Connected devices have made autopilot (vehicle without hands on control) and auto parking a reality. Audi and Tesla have embedded "Google Street View" in their navigation system, helping the user to know the traffic conditions ahead. The technology gives the user options to choose a different route and therefore reduce traffic.

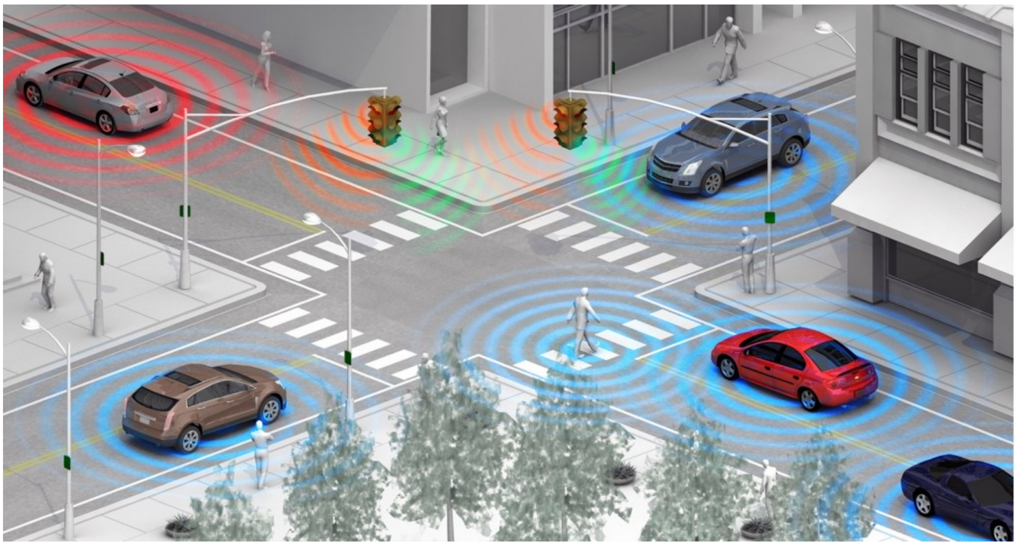

Figure 5 – Example of Smart Connected Car

2.2.7 Industry 4.0

IoT is poised for rapid growth across a wide variety of industries that will connect the physical and the digital world. Participation of major blue chip and networking companies, such as IBM, Intel, Qualcomm, ARM and Cisco, in the IoT technologies is a key validation of its business potential to become a multi-trillion-dollar market. Social networking giant, Facebook is moving towards IoT connectivity by supplying a developer kit for Parse - Facebook backend platform, which will enable developers to create IoT-enabled apps to support companies such as Intel and Broadcom. This kit also enables users and businesses to create apps and share data through connected devices with ease in order to accelerate the delivery of new IoT services at low cost to the market without the Capital Expenditure (CAPEX) and Operating Expense (OPEX) overhead of an IoT backend platform.

In 2016, Atmel Corporation [25] introduced an automated beer-brewing machine titled ‘Artbrew’ that helps to solve a problem that has challenged craft beer makers for many years. The machine starts brewing automatically with a touch of a button and depending on your choice of beer, it takes one to three weeks to mesh, filter and ferment. The unit uses smart sensor technologies and its app to update the user on the status of the process. This smart technology makes the brewing process accurate and stable, giving the user the most consistent taste. IoT has made operations smarter in multiple ways; businesses have integrated IoT-based technology into their processes to work efficiently. GE is using numerous sensors to measure temperature, air pressure and operational data in real time in their battery plant in New York. This helps the company to monitor their production and processes in real time and also track battery performance. The benefits are higher product quality (less faulty battery), command premium pricing in the market, more production volume per plant, improved employee productivity and a greater customer satisfaction. Major chip companies and networking companies have taken one step forward by investing heavily in IoT. In 2015, companies are investing in cloud, data centers and big data analytics, IBM had announced an investment of $3 billion over the next four years into a separate IoT division. Cisco acquired Jasper Wireless for $1.4 billion, a market leader with a cellular connectivity software platform and also acquired a cloud security company ‘Open DNS’ for $635 million to enhance its position in the IoT market [26]. IoT has also transformed the healthcare industry by providing increased efficiency, lower costs and enhanced patient care. IoT is the driver for revolutionizing patient care as quoted in the latest report by Mind Commerce entitled “Big Data in IoT” it is stated that ‘The use of IoT is expected to grow fastest in healthcare over the next few years, to the tune of $117 billion by 2020’ [27]. Through connected devices, one can capture real-time data and analyze it, thus helping healthcare providers maintain compliance and conduct research for enhanced patient care. IoT is being adopted across industries, such as transportation and logistics (T&L). The sector is increasingly using smart devices to capture and share data to enable greater visibility and understanding of the logistics of transportation. Companies can now pinpoint where trucks are at any moment in time, what delivery has been made and locate missing or late vehicles in emergency situations. A recent Forrester [28] study reveals that 96 percent of T&L companies agree that the IoT is the most strategic technological initiative their organization will undertake this decade. Security sensors, Wi-Fi and real-time locating systems are the technologies that will help drive the global adoption.

2.3 IoT communication protocols

This section analyzes the main protocols used for IoT at different levels of the network, especially at the MAC, Network, Transport, and Application layer.

2.3.1 MAC Protocols

An IoT application must specify how the various objects are interconnected starting from the bottom of the TCP/IP Stack.

The most used MAC protocols for the IoT are 802.15.4 [29], WiFi [30] and Bluetooth Low Energy [31].

IEEE 802.15.4 is the most widespread protocol for Wireless Sensor Networks (WSN). In 802.15.4 the nodes connect to a coordinator in a star, tree or mesh topology. The coordinator might be application specific or provide Internet connectivity to the nodes.

Estimating application latency in 802.15.4 networks is application specific, because the packets to be received are polled by the nodes at specific intervals. As most nodes are battery-operated, they might sleep for hours before turning on the radio again. However, the latency for sending such message is in the order of 2.4 ms and 6.02 ms, plus any retransmission needed in case of errors.

In [32], the authors have measured that low-power Wi-Fi (LP-WiFi) provides a significant improvement over typical Wi-Fi on both latency and energy consumption counts. According to the authors, LP-Wifi consumes approximately the same power as 802.15.4 for small packets but it performs better for large packets. Thus, it is possible that a LP-Wifi approach will emerge as a solution in some sensor network applications.

Since its first version, Bluetooth has been used to control from cars to wristbands and in general most of our personal devices. Bluetooth imposes a star topology to networked things, placing the user-controlled device as the center. Bluetooth has a very short range, and it is suitable for Personal Area

Network devices. As an example, Bluetooth is used for headsets, speakers, printers, and quantified self devices.

Low Bluetooth Energy (BLE) offers a low power alternative to the standard Bluetooth, reducing latency to 6ms and application throughput to 236.7 kbit/s [33]. However, it reduces the power consumption while connected to 0.024 mA, giving an expected battery life of 1 year over a coin battery. BLE chips are cheaper than 802.15.4 chip, but it requires more processing power.

2.3.2 Network and Transport Protocols

The network and transport layers of any Internet of Things applications are extremely important to achieve interoperability between different solutions. On one hand, the user is always connected through the Internet stack, which right now involve the Internet Protocol (IPv4) [34] as network protocol and the Transmission Control Protocol (TCP) [35] as transport protocol. On the other hand, the things might be connected through different protocols, such as ZigBee.

At the network level, the major issue of IPv4 is its byte address field length, which is only 32 bits. As of today, all the possible addresses are allocated. Therefore, the next version of the Internet Protocol, called IPv6, uses 128 bits for its address field, which allows plenty of addresses for all the possible things. However, IPv6 headers are much larger than IPv4, and the IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) standard specifies how to compress them to fit in a 802.15.4 frame.

The TCP creates a communication channel between two remote parties, a client and a server. TCP is the basis of the World Wide Web, as it creates a reliable communication channel between the parties by involving retransmissions. In [4], the authors discuss the reasons why TCP is not sufficient as a transport protocol for the IoT: connection setup, congestion setup, and data buffering makes TCP expensive to send end-to-end messages on battery-powered devices that are in a sleeping state. Thus, TCP cannot be used on sensors that have an estimated battery lifetime of years.

The User Datagram Protocol (UDP) [36] offers the minimum set of features for a transport protocol: application multiplexing, via port numbers, and integrity verification, via checksum, of the header and payload. The main difference with TCP is that UDP is not reliable: the application is responsible for handling the retransmissions of lost messages. Thus, IoT applications can customize the trade-off between reliability, congestion control, and battery consumption.

ZigBee is a network, and application protocol suite that aims to solve the industry and home automation problem. Thus, it is not compatible with the Internet stack, and requires a gateway. ZigBee uses 802.15.4 as its MAC layer, thus it supports start, tree, and mesh topologies. At the network level, ZigBee supports network routing through the Ad hoc on-demand distance vector (AODV) routing algorithm [37].

The ZigBee standard includes no transport layer, but it have several application profiles that specifies the functionalities of the things: these profiles dictates the available data across different vendors.

2.3.3 Application Protocols

At the application layer, thing-driven approaches leverage binary protocols and data formats that are specifically designed for machine to machine communications. These protocols and data format introduce little overhead, minimize battery consumption but are usually not reused in other fields. The benchmark against which all these protocols should measure is HTTP, as it is extremely familiar to the developers.

The most widespread open protocols specifically designed for the IoT are MQTT [38] and the Constrained Application Protocol (CoAP) [39], which are based on TCP and UDP, respectively. MQTT is a classical publish/subscribe protocol, while CoAP is a request/response protocol based on the REST pattern. Both MQTT and CoAP support the same primitives: MQTT focuses on sending and receiving updates, while sup- porting basic syndication; CoAP focuses on syndication, while supporting a basic notification mechanism.

Hypertext Transfer Protocol (HTTP) [40] is the basis of the Web, and it is used also to integrate different software applications using the Representational State Transfer pattern [41], where every resource is globally identified by an Uniform Resource Identifier (URI). As of today thousands of businesses offer REST APIs for creating new applications. It is also important to note that HTTP is a text-based protocol with many data types being transferred in text format. HTTP is designed to support caching and several approaches exist to syndicate data.

The MQTT protocol is fast, lightweight, power efficient and implements various levels of Quality of Service (QoS). MQTT is based on TCP, so it provides standard TCP delivery reliability, in addition to its own QoS mechanism. MQTT implements a classic publish/subscribe (pub- /sub) pattern with a central broker. The protocol revolves around the concept of topic, where clients might publish updates or subscribe to for receiving the updates from other clients. The MQTT community claims that a pub/sub protocol is what is needed to build a true IoT. MQTT can also tunneled over a Web Socket, thus allowing web client to communicate with the nodes with extremely low latency.

The Constrained Application Protocol (CoAP) is an implementation of the Representational State Transfer pattern (REST) and it is similar in HTTP from a high-level point of view. However, it is implemented over UDP and it is binary. Thus, it significantly reduces the overhead for battery-powered devices while guaranteeing HTTP compatibility through a proxy. CoAP supports a basic notification mechanism, the observe option, which is similar to the HTTP Server-Sent Events.

2.4 Security in IoT

Due to the global connectivity and sensitivity of applications, security in real deployments in the IoT is a fundamental requirement.

Following are some security services needed for the IoT:

Confidentiality: Messages that flow between a source and a destination could be easily intercepted by an attacker and secret contents are revealed. Therefore, these messages should be hidden from the intermediate entities; in other words, End-to-End (E2E) message secrecy is required in the IoT. Also, the stored data inside an IoT device should be hidden from unauthorized entities. Confidentiality services ensure this through encryption/decryption.

Data Integrity: No intermediary between a source and a destination should be able to undetectably change secret contents of messages, for example a medical data of a patient. Also, stored data should not be undetectably modified. Message Integrity Codes (MIC) are mostly used to provide this service.

Source Integrity or Authentication: Communicating end points should be able to verify the identities of each other to ensure that they are communicating with the entities who they claim to be. Different authentication schemes exist.

Availability: For smooth working of the IoT and access to data whenever needed, it is also important that services that applications offer should be always available and work properly. In other words, intrusions and malicious activities should be detected. Intrusion Detection Systems (IDSs) and firewalls, in addition to the security mechanisms above, are used to ensure availability security services.

Replay Protection: Last but not least, a compromised intermediate node can store a data packet and replay it at later stage. The replayed packet can contain a typical sensor reading (e.g. a temperature reading) or a paid service request. It is therefore important that there should be mechanisms to detect duplicate or replayed messages. Replay protection or freshness security services provide this, which can be achieved through integrity-protected timestamps, sequence numbers, etc.

3 Moving Computation and Services to the edges

3.1 Literature Review

The first concept of edge computing dates back to the 1990s, with the introduction of content delivery network (CDNs) by Akamai to accelerate web performance [42]. A CDN uses nodes at the edge close to end users to perform prefetching and caching techniques for the web content.

In 2012, Flavio Bonomi et. al presented one of the first works on Fog Computing assessing the suitability for the IoT. [20]. In this work the author presents the requirements of emerging applications in terms of location awareness, real time interactions and need for geo-distributed end-points and how Fog Computing addresses these issues. Furthermore, they provide further insight into the suitability of Fog Commuting for IoT applications with a few use-cases including a smart traffic light system and a smart wind farm in the following paper [21].

MEC aims at reducing network stress by shifting computational efforts from the Internet to the mobile edge.

The MEC concept has mainly been discussed from a non- technical perspective by IBM which discusses economical benefits for businesses and M2M applications [43] and Nokia Networks which introduce a first real-world MEC platform [44]. In this approach, MEC servers are standard IT equipment with processing and storage capacity directly placed at mobile network's base stations, which are capable of collecting real- time network data like cell congestion, subscriber locations, and movement directions.

The most important advantage of edge computing inside cellular networks is given by a reduction of end-to-end delay. Moreover, storing only relevant information for the coverage of a single cell implies a reduction in computational overhead since many tasks can be performed on smaller data sets.

There are some related approaches that are similar to the MEC concept.

Therefore, Mobile Cloud Computing (MCC) is highly related to mobile edge computing. In [45] is presented a survey on Mobile Cloud Computing describing cloud-affine mobile application types. The Cloudlets are an example of MCC [46] **.**

The main features of Cloudlets are: resource wealth, fast and stable Internet access and provide computing, bandwidth, and storage resources to nearby mobile users.

While cloudlets are owned and managed by mobile end users, MEC servers are operated by mobile infrastructure provider. Being co-located with base stations, MEC servers provide additional features such as being able to access position and mobility information.

3.2 Fog Computing

3.2.1 Overview

As defined in [47], Cloud Computing can be thought as an aggregation of Computing, providing on demand network access to computing, configurable and shared resources, which can be rapidly supplied and released with minimal management effort or service provider interaction. A key requirement for a Cloud provider is the virtualization of resources in order to allow the required scalability of the Cloud, giving the users a perception of infinite resources [48], providing high storage capacity, high flexibility and high-computing performance [49].

Today we are going to the fourth industrial revolution related to the Internet of Things that consists of having an increasing amount of objects connected to the Internet with their own IP address, to exchange useful data to be processed and analyzed.

However, the transmission of all these data to the Cloud might be inefficient because of requiring high processing capacity and bandwidth, increasing the latencies and consequently negatively affecting the performance of the entire network and even leading technologies such as optical fiber or 4G connectivity having limitations caused by the cost of traffic and availability of bandwidth.

To support the Cloud, by a meteorological metaphor, originates “Fog Computing”, an innovative paradigm which is an extension of the Cloud with the basic idea to transfer a part of processing to the edge of network, close to the end-users in order to solve some problems related to Internet of Things, e.g. the availability of bandwidth and the network’s latencies.

Whereas Cloud Computing is based on large data centers away from the user, the Fog promises to bring more processing power in the network edge [21]. This is even more important for the devices themselves or local gateways, reducing the amount of data to be transmitted to the Cloud, allowing a highly virtualized platform in order to provide processing, analysis, storage, and networking services between end devices and data centers, other than supporting large-scale of sensor networks [20].

In Fog Computing paradigm any irrelevant information is filtered and discarded, whereas the most important ones are forwarded. One of the main goals of Fog Computing is to exploit the available resource of the end devices to allow the network to have a better-distributed intelligence and enhance the performance.

In [50] Fog Computing is defined as “a scenario where a huge number of heterogeneous (wireless and sometimes autonomous) ubiquitous and decentralized devices communicate and potentially cooperate among them and with the network to perform storage and processing tasks without the intervention of third-parties. These tasks can be for supporting basic network functions or new services and applications that run in a sandboxed environment.

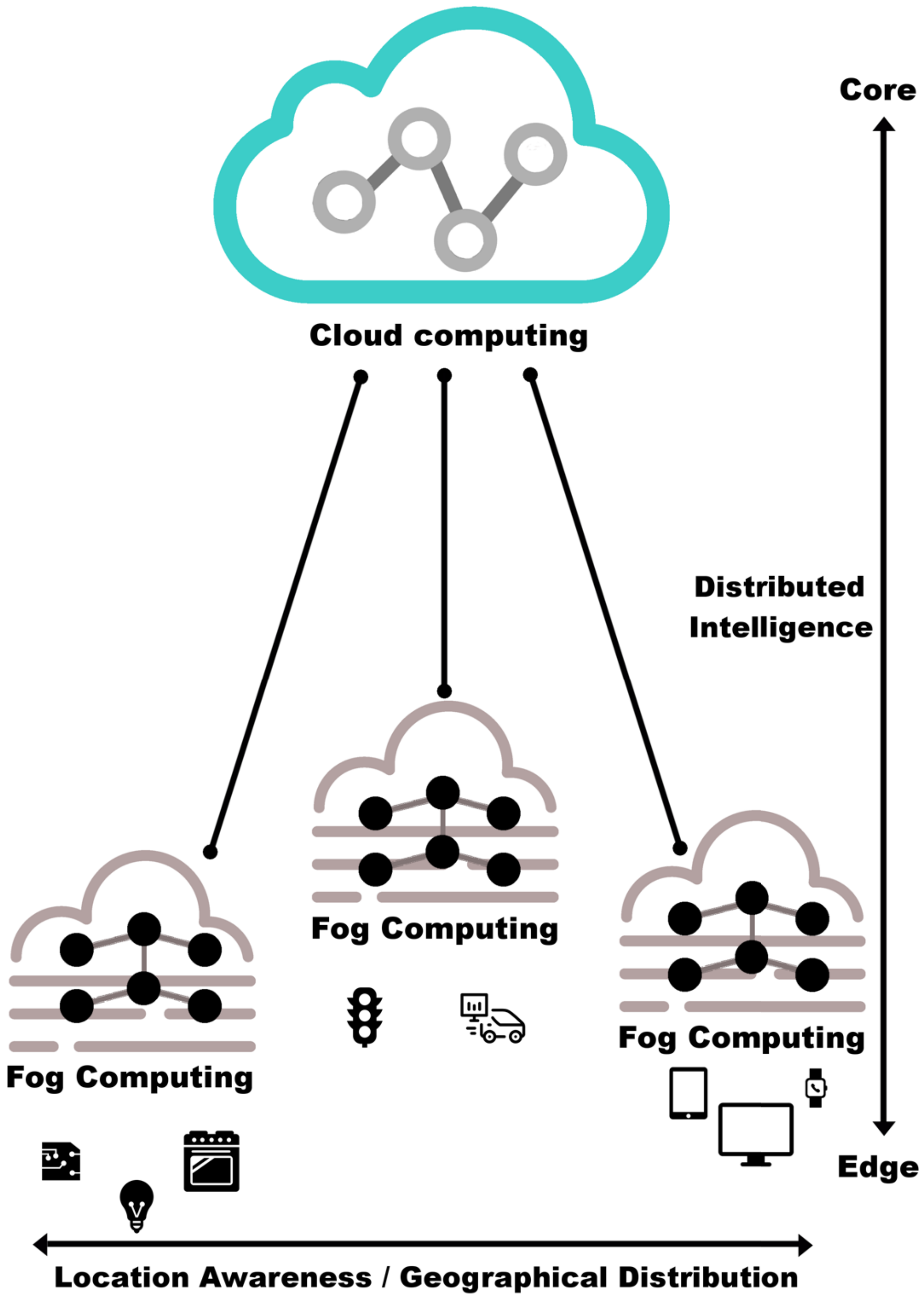

Figure 6 - Fog Computing scenario: the network has a distributed intelligence extended to the edge while the end devices present location awareness and geographical distribution.

3.2.2 Key features

The main features of Fog Computing are:

Location awareness and low latency capabilities: this feature is crucial to support endpoints with rich services to the edge of network including application with low latency requirements.

Geographical distribution: Unlike the centralized Cloud the services and applications deployed in the Fog are widely distributed among several geographic positions.

Support for mobility: Many application need to communicate directly with the mobile devices and the Fog allows to decouple host identity from location identity through mobility techniques such as the LISP protocol [51].

Heterogeneity of resources: Fog nodes come in different form factors, and will be deployed in a wide variety of environments.

Interoperability and federation: In order to support services (such as streaming) the cooperation of different providers is required. Hence, Fog components must be able to interoperate, and services must be federated across domains.

3.2.3 Applications

Fog Computing brings many benefits in various IoT scenarios such as:

Connected Cars: Fog computing is the ideal for the connected vehicles (CV) because real time interactions will make communication between cars, access points and traffic lights as safe and efficient as possible. Video camera that senses an ambulance flashing lights can automatically change street lights to open lanes for the vehicle to pass through traffic. Smart street lights interact locally with sensors and detect presence of pedestrian and bikers, and measure the distance and speed of approaching vehicles. As shown in Figure 7, intelligent lighting turns on once a sensor identifies movement and switches off as traffic passes. Neighboring smart lights serving as Fog devices coordinate to create green traffic wave and send warning signals to approaching vehicles. Wireless access points like Wi-Fi, 3G, road-side units and smart traffic lights are deployed along the roads. Vehicles-to-Vehicle, vehicle to access points, and access points to access points interactions enrich the application of this scenario [52] [53].

Figure 7 – Example of smart street light

Smart Grids: Fog computing allow fast, machine to machine (M2M) handshakes and human to machine interactions (HMI) which would work in cooperation with the cloud. Energy load balancing applications may run on network edge devices, such as smart meters and micro-grids Based on energy demand, availability and the lowest price, these devices automatically switch to alternative energies like solar and wind. As shown in Figure 4, Fog collectors at the edge process the data generated by grid sensors and devices, and issue control commands to the actuators. They also filter the data to be consumed locally, and send the rest to the higher tiers for visualization, real-time reports and transactional analytics. Fog supports ephemeral storage at the lowest tier to semi-permanent storage at the highest tier. Global coverage is provided by the Cloud with business intelligence analytics [54].

Wireless Sensor and Actuator Networks: Traditional wireless sensor networks fall short in applications that go beyond sensing and tracking, but require actuators to exert physical actions like opening, closing or even carrying sensors. In this scenario, actuators serving as Fog devices can control the measurement process itself, the stability and the oscillatory behaviors by creating a closed-loop system. For example, in the scenario of self-maintaining trains, sensor monitoring on a train’s ball-bearing can detect heat levels, allowing applications to send an automatic alert to the train operator to stop the train at next station for emergency maintenance and avoid potential derailment. In lifesaving air vents scenario, sensors on vents monitor air conditions flowing in and out of mines and automatically change air-flow if conditions become dangerous to miners.

Smart Cities: Fog computing would be able to obtain sensor data on all levels of activities of cities and integrate all the mutually independent network entities within. The applications of this scenario are facilitated by wireless sensors deployed to measure temperature, humidity, or levels of various gases in the building atmosphere. In this case, information can be exchanged among all sensors in a floor, and their readings can be combined to form reliable measurements. Sensors will use distributed decision making and activation at Fog devices to react to data. The system components may then work together to lower the temperature, inject fresh air or open windows. Air conditioners can remove moisture from the air or increase the humidity. Sensors can also trace and react to movements (e.g., by turning light on or off). Fog devices could be assigned at each floor and could collaborate on higher level of actuation. With Fog computing applied in this scenario, smart buildings can maintain their fabric, external and internal environments to conserve energy, water and other resources [55].

3.3 Mobile Edge Computing

3.3.1 Overview

The concept of Mobile Edge Computing (MEC) aims to create a standardized and open environment, providing IT capabilities such as computational resources, storage capabilities, connectivity and access to user traffic and network information within the Radio Access Network (RAN) in close proximity to mobile subscribers [56].

Mobile-edge Computing allows content, services and applications to be accelerated, increasing responsiveness from the edge. The mobile subscriber's experience can be enriched through efficient network and service operations, based on insight into the radio and network conditions.

Operators can open the radio network edge to third-party partners, allowing them to rapidly deploy innovative applications and services towards mobile subscribers, enterprises and other vertical segments. Proximity, context, agility and speed can be translated into value and can create opportunities for mobile operators, service and content providers, Over the Top (OTT) players and Independent Software Vendors (ISVs), enabling them to play complementary and profitable roles within their respective business models and allowing them to monetize the mobile broadband experience.

This environment can create a new value chain and an energized ecosystem comprising application developers, content providers, OTT players, network equipment vendors and mobile operators. Based on innovation and business value, this value chain will allow all players to benefit from greater cooperation.

3.3.2 MEC Architecture

The key element of this architecture is composed of a MEC IT application server that can be directly integrated to the LTE access node level (e.g. ENB), or at the level of the concentrator in a so-called "aggregation site" (3G networks / LTE).

In [56] the MEC server is explained as shown in Figure 8. It is composed by MEC application platform and MEC hosting infrastructure:

MEC application platform layer provides the capabilities to host applications and it is composed by the application's virtualization manager and the application platform services. The first one provides an Infrastructure as a Service (IaaS) structure supporting a flexible and efficient, multi-tenancy, run-time and hosting environment for different kinds of applications. The application platform provides a set of middleware application services and infrastructure services to the applications hosted on the MEC platform. MEC applications from vendors, service providers, and third-parties are deployed and executed within Virtual Machines.

The IaaS controller provides a security and resource sandbox for the applications and the platform. Virtual-appliance applications run on top of an IaaS and are delivered as packaged operating system Virtual Machine (VM) images, allowing complete freedom of implementation.

The MEC application-platform services provide the following set of middleware services to the applications which are hosted on the MEC server:

- Infrastructure services consists of Communication services and Service registry. The communication services allow applications hosted on a single MEC server to communicate with the application-platform services (through well-defined Application Programming Interfaces (APIs)) and with each other (through a service-specific API). The service registry provides visibility of the services available on the MEC server. It uses the concept of loose coupling of services, providing flexibility in application deployment. In addition, the service registry presents service availability (status of the service) together with the related interfaces and versions. It is used by applications to discover and locate the end-points for the services they require, and to publish their own service end-point for other applications to use. The access to the service registry is controlled (authenticated and authorized).

- Radio Network Information Services (RNIS provide indications relating to the activation of User Equipment (UE) on a specific mobile network element. These include parameters on the UE context and the established E-UTRAN Radio Access Bearer (E-RAB), such as QoS, Cell ID for the E-RAB, identity of the UE-associated logical signaling connection, etc.
- Traffic Offload Function (TOF) service prioritizes traffic and routes the selected, policy-based, user-data stream to and from applications that are authorized to receive the data

MEC hosting infrastructure layer is loosely coupled from the applications. It is composed of hardware resources and a virtualization layer and includes the connectivity to the radio network element (eNB or RNC). Multiple implementation options can be used to integrate the server within the RAN. MEC is able to run software locally isolating it from the rest of the network. This feature is especially important in Machine-To-Machine scenarios, for example when it comes to security systems to require high levels of reliability. The proximity makes it possible to access directly to the devices, which can be easily exploited for specific applications.

Other features are lower latency, because the services are performed at the edge of the network close to end devices, and location awareness that allows to get a whole range of business-oriented use cases, including location services, analytics, etc.

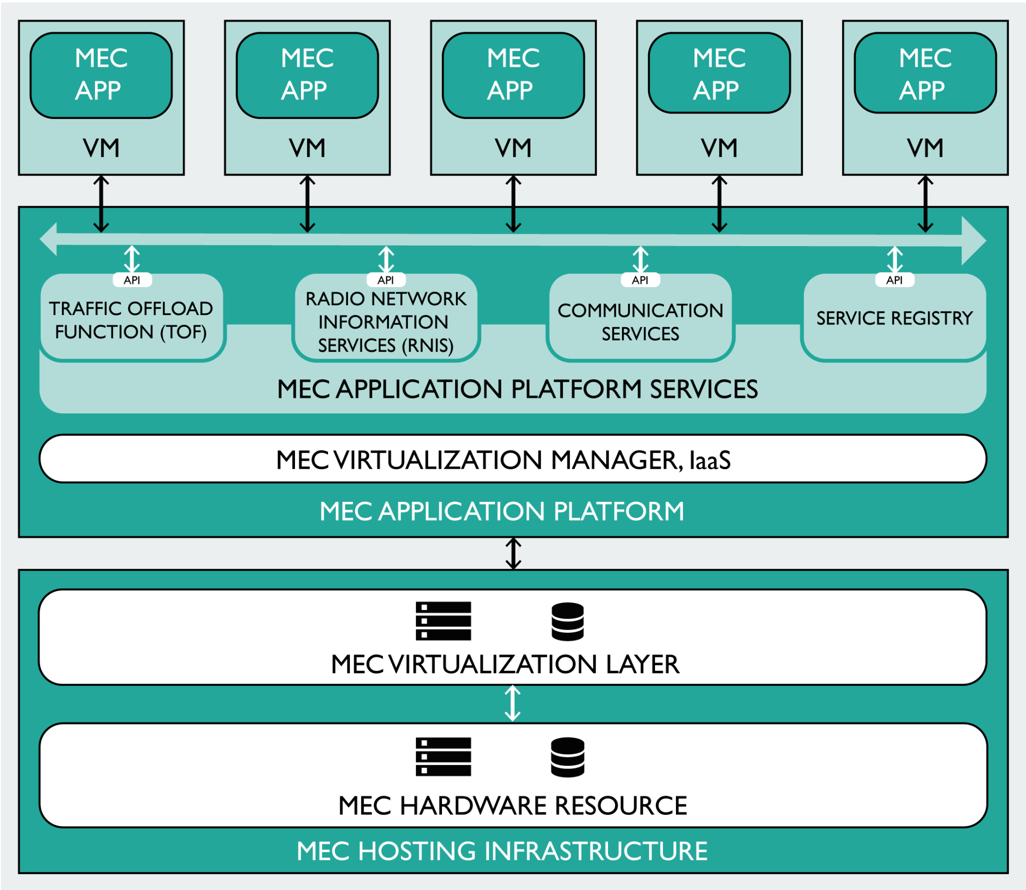

Figure 8 – MEC Server Architecture

Mobile Edge Computing can be categorized by the following properties [57]:

- *Proximity:* In Mobile Edge Computing, edge network is accessed by the mobile devices using RAN. Mobile or portable devices can also connect to the nearby devices through device to device (D2D) communication and simultaneously mobile devices can access edge server located at the mobile base station. Since edge server is nearby to devices, it can extract device information and analyze user's behavior to improve services.

- *Dense Geographical Distribution:* Mobile Edge Computing host IT and cloud computing services at the edge network which sits at numerous locations. Dense geographical dispersed infrastructure contributes in many ways. Services can be provided based on user mobility without traversing the entire WAN.
- *Low Latency:* One of the goals of Mobile Edge Computing is to reduce latency when accessing the core cloud. In Mobile Edge Computing, applications are hosted at the Mobile Edge sever or cloud located at the edge network. Since the available bandwidth within the edge network is high in compare to the core network, average network latency is reduced.
- *Location Awareness:* As the mobile devices are at the close proximity of the edge network, base station collects user's mobility pattern and predict the future network status. Application developers uses user location to provide context-aware services to the user.

- *Network Context Information:* Real time RAN information (such as subscriber location, radio condition, network load *etc.*) are used to provide context related services to the mobile subscriber. RAN information are used by the application developers and content provider to service providers of services, thus improving user satisfaction and Quality-of-Experience(QoE).

3.3.3 **Applications**

There are many potential applications that exploit the Mobile Edge Computing Architecture:

Smart Building Services: Much of the data generated in smart buildings is inherently local and involves device to device communication that could be processed by a locally-hosted IoT gateway. Examples of services that can benefit from local computing and control capability include security, tracking, climate control, smart signage, entry-control, etc.

Internet of Things (IoT): MEC can be used to process and aggregate the small packets generated by IoT services before they reach the core net- work. This will be important for scalability as the number of IoT connections increase and may be crucial for battery-powered IoT devices. A shorter transmission time between device and application server reduces drain on the battery and, therefore, can increase the life of the device and improve the business case for the service.

Vehicle Services: Vehicle-to-infrastructure (V2I) use cases require local processing and low latency, which can be offered by MEC. As such, they are also an example of how MEC concepts are expected to be important to 5G services. This may extend to the ultra-low latency requirements of autonomous vehicles and other mission critical applications of the future.

Augmented Reality: To overlay information from the phone camera (or, perhaps, eyewear in the future), requires localized content to be rendered very quickly on the viewing surface. Ergo, it would benefit from local processing and is an ideal candidate for MEC.

4 Application and Proof of Concept Implementation

This section discusses the work done during the three years of the PhD program regarding my research activity on edge computing. For some of these projects, scientific papers have been produced in collaboration with TIM Joint Open Lab WAVE researchers and University of Catania.

4.1 A Fog computing based architecture for precision agriculture

4.1.1 Scenario

In this work, we focused on a particular application area represented by precision agriculture which, in recent years, is expanding rapidly to become an essential concept in IoT scenario, due to the increasingly number of sensors and connected objects, which is estimated to grow even more rapidly. In fact, in 2050 the world population will reach 9 billion and, as a consequence, it will increase also the food production [58], [59]. Precision agriculture includes all the techniques of farmland management, taking into account the inherent and induced soil variability and the specific needs of crops in order to increase production, minimize environmental damage and raising qualitative standards of the agricultural products [60] [61].

The research study behind this work is to deal with the new challenges of IoT related to the computational loads due to the huge number of devices. For this reason, we propose a novel architectural model based on Fog Computing paradigm to transfer a part of processing abilities both to the gateway and sensor nodes, reducing the computational load of the Cloud and improving scalability, Quality of Service (QoS) and the performance of entire network [62]. This framework results also more insightful looking at futures scenarios in which, not only the agriculture but also many vertical areas, will be composed of thousands of sensors that continuously transmit large amounts of data to the Cloud platforms. In terms of connectivity, we exploit new communication technologies suitable for IoT, named LoRa (Long Range) [63] [64], a type of connectivity interposed between short-range multi-hop technologies, operating in the unlicensed frequency bands, and long-range cellular-based solutions which use licensed broadband cellular standards, named Low-Power Wide Area Networks (LPWANs) [65].

Therefore, we propose an architectural model based on Fog Computing paradigm exploiting the full potential and resources of the peripheral devices considering two Fog layers.

The main services provided at the Fog layer will be the use of clustering algorithms to identify homogeneous areas in order to help the user to manage the entire agricultural land, forecasting analysis to predict possible diseases that can affect the plantations and the alert management to detect abnormal events.

We also simulated and compared the data traffic sent to the Cloud and data storage in both cases with a Fog Computing approach and with a traditional Cloud approach, without exploiting Fog.

Finally, we describe our prototype implemented with the aim to provide a useful tool to manage agricultural environments, such as fields and greenhouses with the goal to optimize the crop, reducing costs and environmental impact.

4.1.2 Precision agriculture

The term "Precision Agriculture" (PA) indicates a group of concepts of agronomic management based on observation and response to variations that exist within growing areas (e.g. soil, moisture, organic matter, etc.) and actions aimed at optimizing the crop.

In an agricultural land we can address soil conditions, weather, sun exposure, and topography very different among them. In the case of small sized fields, farmers may manually and easily change treatments in different areas. However, with the enlargement of the fields and the intensive agricultural mechanization, it has become increasingly difficult to take into account the variability of the field without a revolutionary development in information technology [66].

In agriculture, any plot is characterized by a certain variability involving all the parameters of the soil. The variability observed in the field is the result of interaction between a spatial component and a time component [67]. Thus, we can distinguish between two different concepts of variability, in space and time respectively.

The spatial variability is the ability of a given parameter to occur with a different intensity in the various areas, oscillating around the measurable average value inside them.

The temporal variability can be defined as the ability of a given parameter to assume a different intensity over the time at the same point within the area.

The study of the spatial and temporal variability aims at identifying and quantifying the intensity of one or more parameters and to characterize its main components.

The final target is to determine plots of areas, which are stable over the time, and identify individual parts of them where the productivity factors are different from the other areas. One of the most ambitious and interesting aspects that emerge from PA is, therefore, an attempt to combine two apparently divergent goals: maximize productivity by reducing both the environmental and economic costs.

To pursue this target, a detailed knowledge of cultivation parameters, topographic and weather-environmental is required. Therefore, fertilization and irrigation are two crucial moments, which involve the cultivation process, and optimizing these steps can be a useful tool to improve quality and quantity of the crop yields, also reducing the environmental impact.

By an economic perspective, the Information and Communication Technologies (ICTs) for PA provides the farmers with the possibility to change the distribution and timing of fertilizers according to the spatial and temporal variability of field. Thus, it will be possible to carry out economic analyses based on the variability of crop yield, getting an accurate risk assessment.

Further economic savings may come from the appropriate use of the irrigation of the fields in fact, dosing correctly the amount of water in the various areas will produce a considerable water saving.

With regards to the environmental aspect, precision agriculture represents a smart tool to achieve a precise and targeted use of fertilizer in such a way as to have a substantial reduction of the use of chemicals.

4.1.3 **The Fog-based framework**

As shown in Figure 9, we present a three-tier architecture composed by M2M platform, gateway and sensor nodes. M2M platform represents the Cloud service, able to provide Data Storage, Data Visualization, Network Management and Data Report.

Fog Collector Node (FCN) and Fog Aggregator Node (FAN) compose the underlying layers.

FCN is represented by a gateway, which performs some relevant task, such as clustering analysis, alert and actuation management and forecasting analysis, in order to reduce both important computational load to the Cloud and time response of some events.

FAN is represented by sensor node. It implements an algorithm which performs data filtering and data aggregation processes in order to reduce the amount of data sent to the Cloud, other than the energy consumption.

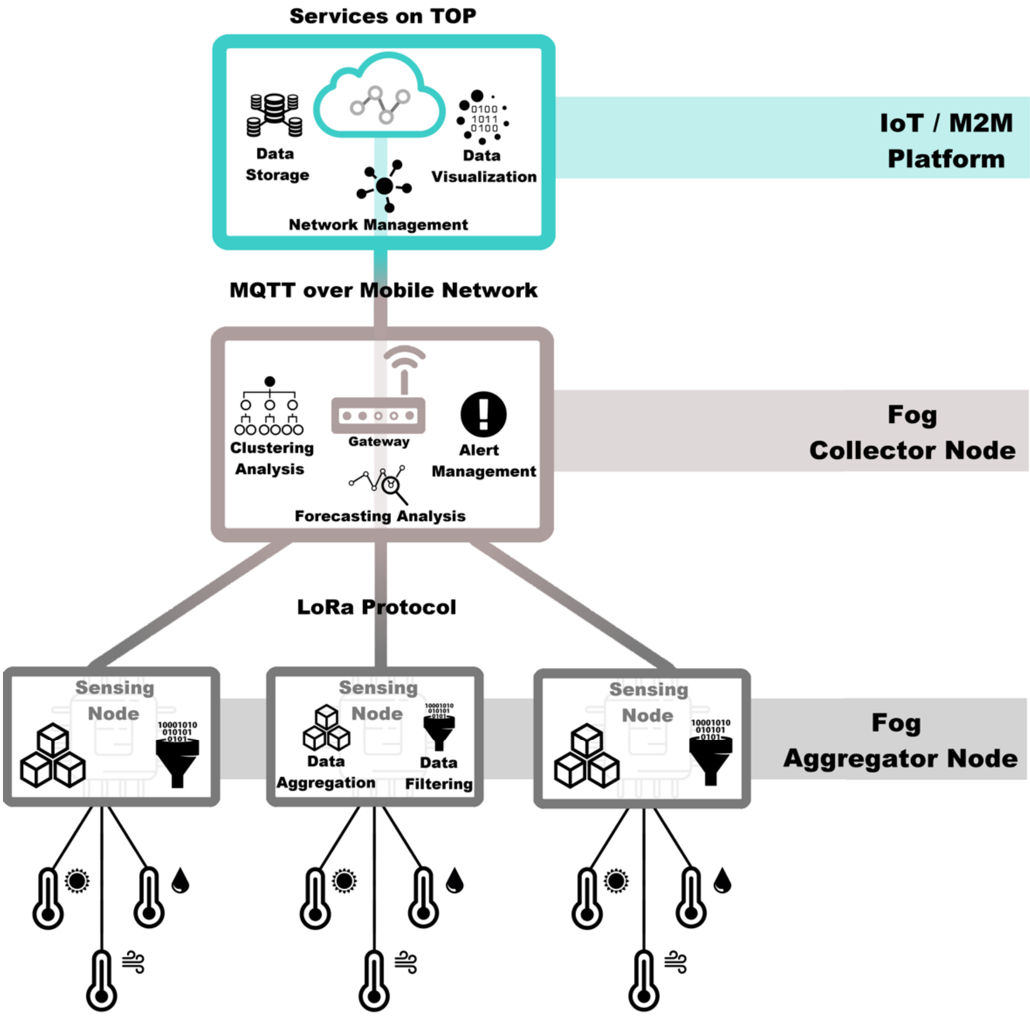

Figure 9 - Proposed framework in precision agriculture scenario exploiting the Fog Computing paradigm

Communication among layers was determined using the efficient protocols and technologies suitable for IoT both saving resources and to reach far away areas.

Currently, the most widely used protocol to connect things is ZigBee, which is an IEEE 802.15.4 standard. Networks exploiting this protocol use a mesh topology and operate mainly in the 2.4 GHz and sometimes in the 868/915 MHz unlicensed frequency bands. Nodes communicating over ZigBee, covering distances ranging from a few meters up to roughly 100 meters, according to the features of the environment [65].

Therefore, to communicate between the sensor node and the gateway we have chosen *LoRa* [63], [64], the innovative wireless technology that allows long distance communication, low bit rate and low power consumption, more suitable for IoT and M2M applications (see Table 1).

TABLE 1
LORA VS ZIGBEE COMPARISON

	LORA	ZIGBEE
Standard	LoRaWAN	IEEE 802.15.4
Frequency-Band	ISM 868 MHz and 915 MHz	ISM 2.4GHz, 868 MHz and 915 MHz
Topology	Star	Mesh
Range	2-5 km (Urban) 10-15 Km (Rural)	10-100m
Data Rate	0.3 - 50 kbps	250 kbps
Battery	Over 10 years	Some year

In this way we can have only one gateway to handle large agricultural lands exploiting a star topology.
Whereas the communication between gateway and M2M platform is via the MQTT protocol (Message Queuing Telemetry Transport) [38], considered one of the most reference standard for IoT communication. MQTT is a messaging protocol for publishing and subscribing suited to working with limited power computing and connectivity of embedded products.
MQTT protocol is less complex than HTTP, having few message types and a lower message size [68]. It supports connections with edge nodes under constrained environments (low-speed wireless access) due to the poor mobile network coverage in some rural areas. Thus, it results more suitable than AMQP protocol [69]. Furthermore, MQTT performs better in the low throughput scenario with a single device and offers lower latency than CoAP protocol [70].
The data to the Cloud will be sent in JSON or XML format over Mobile 4G/3G Network.
FAN is connected to sensors that monitor both the soil (temperature, humidity) and the plants (fruit diameter, leaf wetness). It is programmed to acquire data from the sensors connected to it and send them to the gateway via LoRa technology. Table 2 shows the sensor's accuracy used in the project.
Before transmitting the data, FAN implements a data filtering process in such a way that the values of the data collected by the sensors belong into an acceptable range, according to the design of the sensors.

TABLE 2
SENSOR'S ACCURACY

SENSOR	ACCURACY
Humidity Sensor (808H5V5)	< ±4%RH (25°C, range 30 ~ 80%) < ±6%RH (range 0 ~ 100%)
Temperature Sensor (MCP9700A)	±2°C (range 0°C ~ +70°C) ±4°C (range -40°C ~ +125°C)
Humidity+Temperature Sensor (SHT75)	±0.4°C (range 0°C ~ +70°C), ±4°C (range -40 ~ +125°C) ±1.8%RH
Fruit Diameter Dendrometer	±2μm
Solar Radiation Sensor PAR (SQ-110)	±5%

For example, the humidity values cannot exceed 100%. However, the values, which are out of range, may appear due to various factors (e.g. measurement errors or compromised sensors, faulty sensor values etc.). If the data value is out of range, these defective sensory data are discarded and if these discarded data exceeds a certain threshold, the FAN sends an alert to FCN to notify a potential anomaly for the specific sensor node.

The next step is the Data Aggregation process. One per-hour value will be transmitted to the gateway as result of the average of the previous values or, if the samples present certain variability, also the min and max values of the series will be sent. These values may constitute useful information for a further analysis carried out at the upper levels.

To assess the variability of the samples, we compare the coefficient of variation with a threshold, which usually has a value equal to 0.5.

Therefore, rather than sending the data read at each time by the sensors, the sensor node transmits an aggregate value, reducing the network traffic and hence energy consumption.

Figure 10 shows the algorithm performed within Fog Aggregator Node.

To evaluate the time complexity of the algorithm we can observe that the first part (data filtering process), from lines 5 to 13, involves a constant complexity and it is executed *N* times due to the while loop defined between lines 4 and 14. Thus, the overall time complexity of the data filtering process is *O(N).*

The second part, that is the data aggregation process, which performs the computation of average and standard deviation of the input data, has a *O(n*) complexity. It is noted that $n \leq N$ since the data filtering process can discard some data. The rest of the statements implies a constant complexity.

Algorithm implemented into Fog Aggregator Node

Input

th_df : Data filtering threshold to detect anomaly of sensor nodes;

th_cv: Coefficient of Variation threshold;

N: Number of sensed data;

Data filtering process

while sensed data < N **do**
  if sensed data is out of range of acceptable values **then**
    Discard data;
    Increment counter variable of discarded data;
  else
    Store data in local database;
    Increment counter variable of sensed data;
  end if

  if discarded data > th_df **then**
    Send alert to FCN;
end while

Data aggregation process

Calculate the average μ of the samples;
Calculate the standard deviation σ;
Calculate coefficient of variation $cv = \frac{\mu}{\sigma}$
if $cv < th\_cv$
  Send the average value;
else
  Send the average, min and max values;
end if

Figure 10 - Data filtering and Data aggregation process

Therefore, we can state that the time complexity involved by the entire algorithm, with respect to the number of sensed data, is *O(N).*

FCN implements a middleware able to receive data from FAN by LoRa technology, store the received data in local RDBMS, transmit data to the Cloud in MQTT over mobile 4G/3G network and, periodically, perform the following tasks.

Cluster Analysis: as explained in [71], we use a hierarchical agglomerative clustering to subdivide the field into homogeneous parts. In this way, the user can assess the field variability and exploits it to maximize the crop, e.g. dosing the proper amount of fertilizer or water depending on the clustering of the various areas.

The clustering algorithm is periodically performed and then the results are transmitted to the platform via MQTT protocol in JSON or XML format.

Forecasting Analysis: the collected data and cluster analysis will be exploited by the data mining algorithms to facilitate and enhance the development of predictive models useful for sustainable agricultural management, in order to predict the occurrence of plant diseases and take the most suitable actions. Furthermore, the weather forecast will be used to better manage irrigation and save water.

Alert Management and Actuation: in presence of certain conditions, e.g. the soil moisture level falls below a certain threshold due to high temperatures or because the system does not work appropriately, the actuator, such as automatic irrigation, will be activated. Therefore, FCN will send the actuation command and then will notify the Cloud of the event. In fact, the actuation takes place at the edges of the network close to the field devices, without the need that the command comes from the Cloud.

Figure 11 shows a flow diagram to explain our framework based on Fog Computing paradigm, where is highlighted the communication and the main tasks performed by FAN and FCN.

The proposed framework presents the following benefits:

- *Computational load balancing:* the distribution of business logic among different levels of the architecture allows having a more balanced computational load saving considerable Cloud resources, which will perform less activities despite the more overhead due to data stored among the end devices.
- *Reduction of waiting times:* when a real-time event occurs, it is processed locally, closer to the field devices such as the actuation process management, without necessarily reaching the Cloud but just sending notifications on related actions and sending notifications about the taken actions. Therefore, it will be the gateway to have a responsibility to begin the actuation.
- *Reduction of hardware costs:* bringing business logic at a lower level allows the use of cheaper radio antennas that enable the use of communication mechanisms developed specifically for IoT. Although ensuring a throughput of some Kb, these communication mechanisms do not affect the overall performance of the system, thanks to the thoroughness with which the same intelligence is distributed.

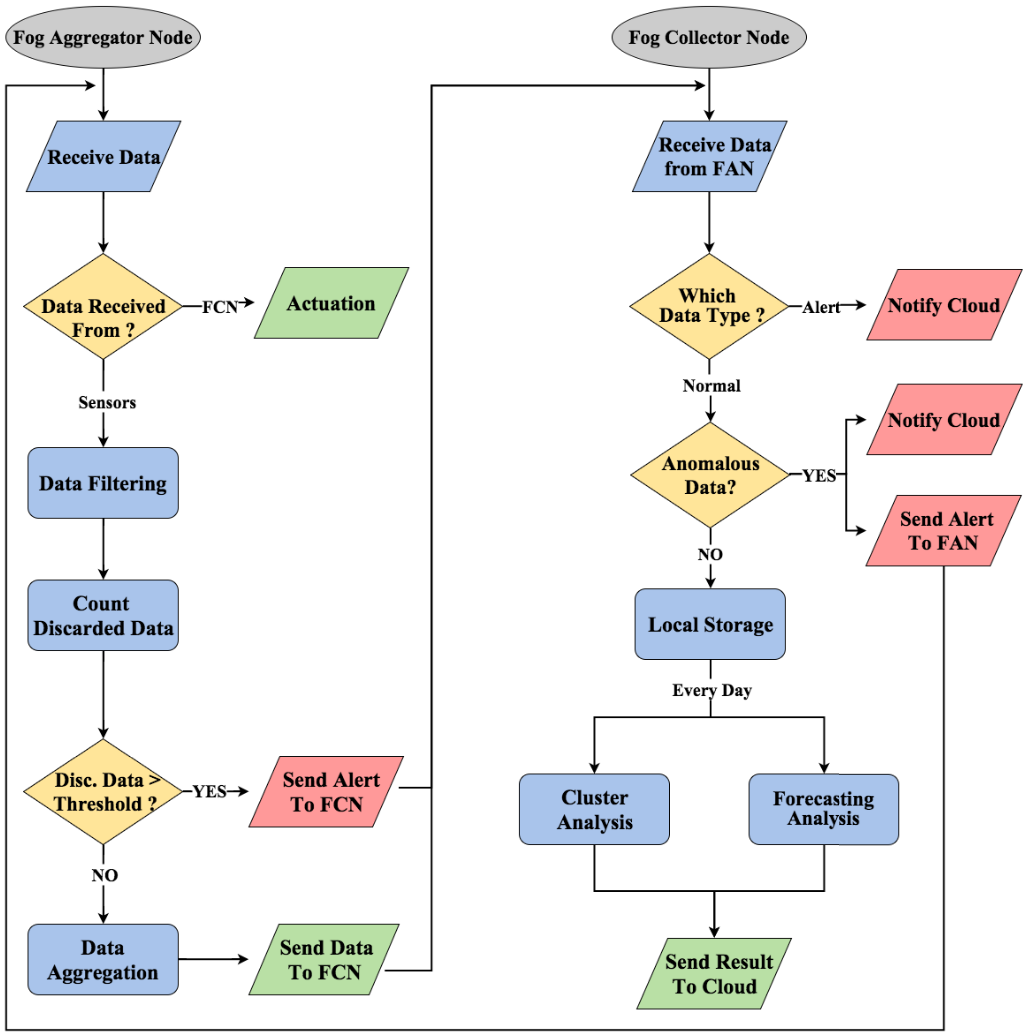

Figure 11 - Flow diagram which describes the Fog logic

4.1.4 Simulation and evaluation of data traffic and data storage

To evaluate the usefulness of the proposed architecture, we carried out two kinds of simulations such as transmitted and stored data to the Cloud.

Our simulations have been performed in a network composed by ten sensor nodes (FAN), the gateway (FCN) and the Cloud platform and we have evaluated the amount of transmitted and stored bytes to the Cloud in three different scenarios:

- Cloud Computing architecture,
- 1-tier Fog architecture,
- 2-tier Fog architecture,

In order to facilitate the evaluation processes, we simulated both gateway and Cloud within the same host.

Table 3 shows some parameters of the data traffic simulation.

TABLE 3
SIMULATION PARAMETERS FOR DATA TRAFFIC

	FROM FAN TO FCN	FROM FCN TO CLOUD
Protocol Type	LoRa	MQTT
Packet Size	110 Bytes	750 Bytes
Data Transmission Frequency	Per-Hour (1-2 tier) Per-Minute (cloud architecture)	Per-Hour (1-2 tier) Per-Minute (cloud architecture)
Devices Number	10 FAN	1 FCN

In 1-tier Fog architecture, the pre-processing is done only in the FAN that senses data from the sensors exploiting the algorithm explained in Figure 10, which then are transmitted to the gateway, transforming them into MQTT format to be sent over the mobile network to the Cloud platform.

In 2-tier Fog architecture, when FCN receives data from FAN, every hour it applies the hierarchical agglomerative clustering algorithm by identifying the homogeneous areas in the field and applying a further data aggregation.

Following the three approaches, Figure 12 compares the amount of transmitted bytes per day, whereas Figure 13 compares the amount of stored data to the Cloud calculated in one month.

The overall quantity of transmitted and stored data to the Cloud is calculated as follows:

$$Bytes_{\,tx\backslash stored} = Packet\backslash Doc_{size} * Min_{day\backslash month} * Devices_{num} \quad (1)$$

$$Bytes_{\,tx\backslash stored} = Packet\backslash Doc_{size} * Hour_{day\backslash month} * Devices_{num} \quad (2)$$

Eq. (1) is used in the Cloud architecture, while Eq. (2) is used in 1-tier and 2-tier architecture. The differences among 1-tier and 2-tier is the devices number, since in 2-tier architecture we consider only one device (gateway) sending an aggregate value as result of cluster analysis.

In both simulations we have used a log-10 scale for y-axis to better show the values by compressing them in a more readable scale.

According to the first graph and exploiting the Fog Computing paradigm, we can observe that the amount of transmitted data presents a sharp reduction in relation to the traditional Cloud Computing architecture.

Exploiting the 2-tier Fog Computing architecture, the amount of transmitted data to the Cloud will be further reduced, missing a part of information respect of the measure of the individual sensor node. Therefore, it will be sent a measurement range for each cluster of nodes since, in this context, it may not be significant to have information on the parameters measured by each node, but it might be enough to have information about overall features of a specific area. To evaluate the stored data in the Cloud, we used a NoSQL database, considering that each stored document related to the normal sensors measures has size of 425 bytes, while each document regarding cluster analysis has a 2 KB size.

Observing the second graph shown in Figure 13, we can see a decrease in the amount of stored data in the Cloud because of the distribution on both the FAN and the FCN. As a consequence, the overall amount of stored data will be higher in 1- or 2-tier approaches, due to data duplication in both the two aggregators.

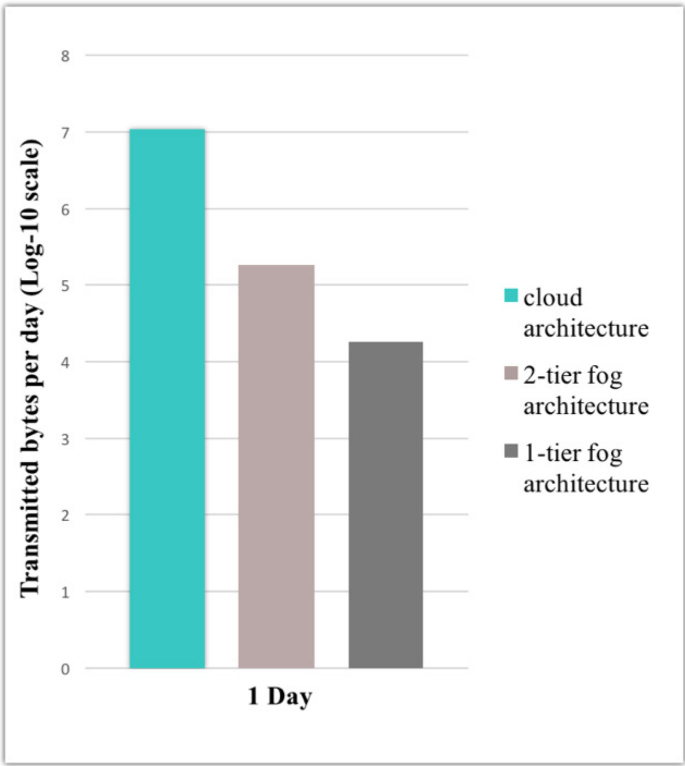

Figure 12 - Data transmitted to the Cloud using the three different approaches: data represented by bytes-per-day in Log-10 scale.

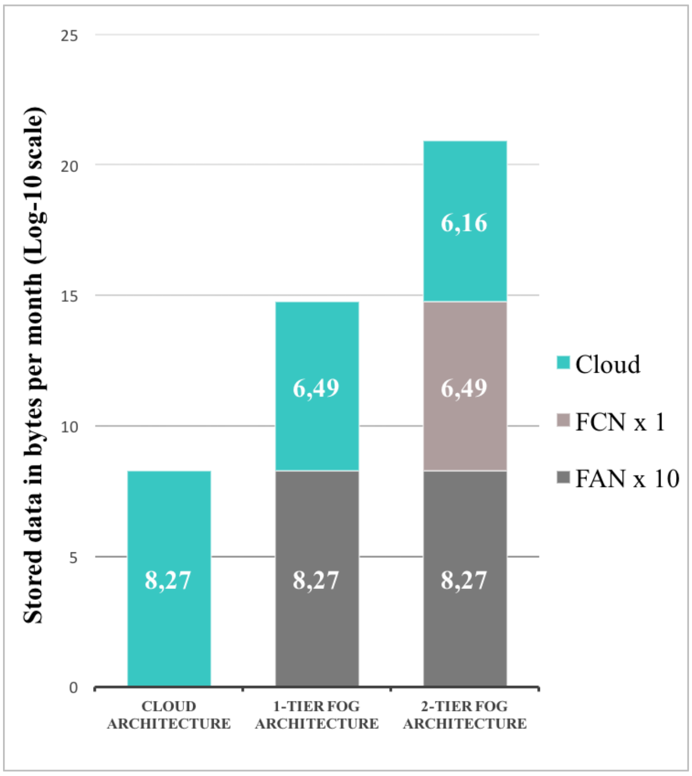

Figure 13 - Data stored in the Cloud using the three different approaches: data represented by bytes-per-month in Log-10 scale.

4.1.5 Prototype

We have developed an application prototype to provide useful services for the optimal management of the agricultural lands. Figure 14 illustrates the subdivision of the field into clusters in order to monitor the field.

The clustering allows identifying homogenous areas that, in this case, are grouped by soil moisture values, exploiting the hierarchical agglomerative clustering. In particular, we can observe three different clusters and the presence of a cluster composed by only SN6 node with a lower soil moisture value.

Figure 15 shows all the alarms produced by some sensor nodes. In fact, SN6 produces an alarm to notify a case where there is a low percentage of humidity. As explained in Figure 11, the FCN will send a command of the additional irrigation to the FAN and will notify the successful operation to the Cloud. The yellow alarm denotes another type of anomaly related to likely faults to the specific sensor nodes due to amount of discarded data during the filtering process. Specifically, we observe potential faults for the humidity sensors of SN5 and SN7, so that the user can perform an accurate control of the specific sensor node and, if necessary, replace it.

Figure 14 - Application prototype to manage agricultural lands: the field is divided into clusters grouped by soil moisture value

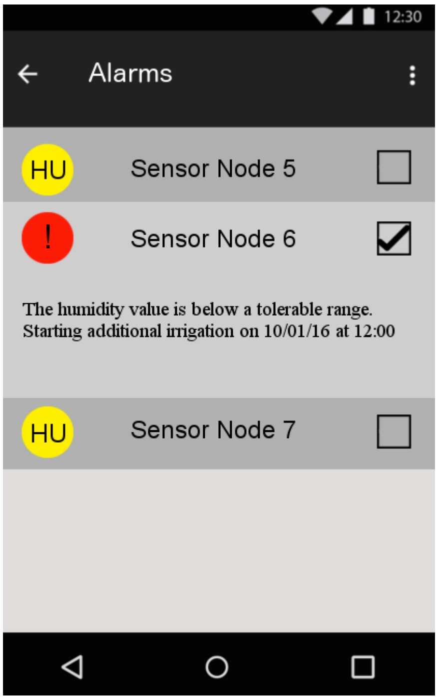

Figure 15 - Application prototype to manage agricultural lands: alarm notification produced by sensor nodes.

4.1.6 Discussion of results and possible developments

In this work we presented a Fog Computing solution in an IoT scenario where the computing is peripherally distributed, balancing the computational load. As application area, we consider the precision agriculture, a field of growing interest in research community, due to its potential impact on performance in terms of management, environmental and economic outcomes. The proposed framework considers two Fog layers, performing respectively different tasks based on their computational capabilities. Thus, FAN nodes are responsible for data filtering and aggregation, whereas clustering analysis, alert and actuation management tasks are carried out by FCN.

Based on this framework, we have simulated and highlighted how the two-tier Fog Computing approach is able to reduce significantly the amount of transmitted data to the Cloud.

Among the achieved benefits exploiting this framework and the implemented prototype, the most valuable are the load balancing, due to the Fog Computing approach, and the reduction of waiting times in the actuation phase of an event. The realized application prototype will allow us managing agricultural lands, thanks to the clustering mechanism, and tracking easily the alarm notifications from the sensor nodes.

Future works will be focused on exploiting data gathered through the prototype and perform data analysis applying data mining algorithms, preventing plants disease and improving the quality of the crop. Moreover, we will work to strengthen the reliability of the measures through the robustness and sensitivity analysis.

4.2 A Mobile Edge Computing solution for smart city services

4.2.1 **Scenario**

In this project we have considered a smart city scenario based on two important services which can improve the quality of life and the security of the citizens. In particular, we focused on smart traffic light system and smart street light automation.

Country roads at some moments are not very busy while working on active lighting. Smart street lighting service enables an innovative light management system for street lighting. It provides demand-oriented control and monitoring for street lamps and permits to create and configure the networks which involve cars and street lamps. The use case exploits this system in order to reduce the energy consumption and turn on the street lamps only when the car requests the service.

The rapid growth in the vehicle ownership is one of the measures for economic growth of country.

However indirect effect of vehicle ownership is acute traffic congestion. The exploitation of new trends and technologies requires fast transportation of goods, machinery and manpower for various reasons [72].

During rush hours, emergency vehicles like Ambulances, Fire Brigade and Police cars are likely to stop in urban traffic. Due to this, these emergency vehicles are not able to reach their destinations in time, resulting into a loss of human lives [73]. In this case, a system used to provide clearance to any emergency vehicle by turning all the red lights to green on the path of the emergency vehicle, hence providing a complete green wave to the desired vehicle can be very useful.

4.2.2 **Proposed solution**

The proposed solution exploits the Mobile Edge Computing paradigm in order to manage and monitor the smart traffic light system and smart street light Automation services in a smart city.

The Mobile Edge Computing is an evolution of the Base Transceiver Station (BTS) and Evolved Node B (eNodeB), providing low latency and high bandwidth services and also allowing direct access to real-time information. All this is guaranteed because part of the processing is performed in the network's edges, close to end users. According to the standard, in the future, eNodeB will be able to host application servers with computational capabilities, storage, connectivity, and user traffic access.

In this context, the development of services for future Smart City, where thousands of smart objects, vehicles, mobiles, people interact among them, finds an important response.

Therefore, this work enables an innovative intelligent lighting service and smart traffic light system managed predominantly at the eNodeB level. In this way, when a user / client on a vehicle requires a service, he will communicate directly with the eNodeB to which he is connected. The eNodeB, after verifying the user identity, will create a virtual local network between the client and one or more service nodes. At this point, the client and service can start an end-to-end communication to activate/deactivate the service.

The service can be customized according to the type of user (e.g.: if an ambulance passes the flash lamps to report an emergency).

The handover option will also be handled, i.e. when a node (user) changes the base station to which it is connected, it will make a new request / update of the service to find out about the new service nodes nearby.

Cloud will be used exclusively for maintenance and monitoring of lighting management objects. Periodically, every eNodeB will send the Cloud fault data and the actual use of the service to Cloud, while all else will only be handled at base station level.

4.2.3 Prototype operation

In this section the operation of the system is explained. A prototype was implemented exploiting both hardware and software technologies.

To emulate client, service and BTS we used open hardware boards such as Arduino and Raspberry Pi, whereas for the communication among devices we used Alljoyn protocol [74]. The entire code of the implemented solution is written in NodeJS [75].

The solution is based on Stack4Things (S4T)**,** an OpenStack-based framework spanning the Infrastructure-as-a-Service and Platform-as-a-Service layers which enables developers and users to manage an IoT infrastructure, remotely controlling nodes as well as virtualizing their functions and creating network overlays among them, implementing a provisioning model for Cyber-Physical Systems [76]. Therefore, S4T will be used to create virtual networks useful for end-to-end communication of the devices.

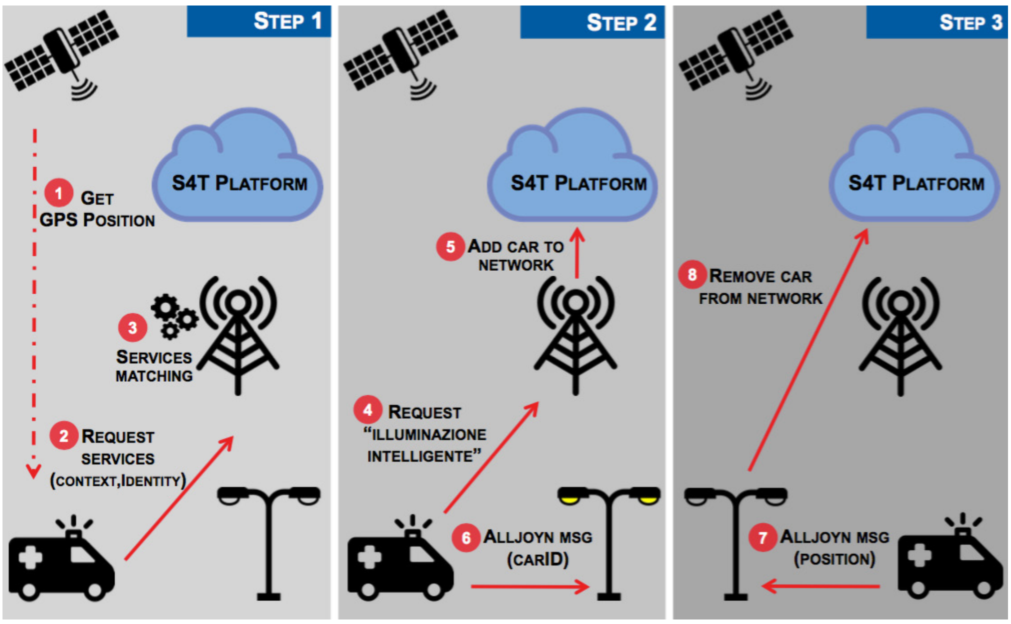

Figure 16 – The operating principle that enables the smart street lighting service

Figure 16 shows the logic that enable the smart street lighting service:

1. The vehicle gets the GPS coordinates from the satellite.
2. The vehicle demands all the services that can exploit (discovery services), sending its context and identity to the controller (BTS), in such a way that the BTS can understand which type of user is making the request (normal user, ambulances, etc.).
3. The controller makes a Services Matching returning all the services to which the user is enabled with the relative positions.
4. When the user (vehicle) realizes to be in proximity to a service, performs a specific request to the BTS.
5. The BTS adds the car to the virtual network of the service manager.
6. Exploiting the Alljoyn protocol, the nodes will communicate with each other and the car will send its id to the Service Manager/Provider to receive the custom service (e.g. if an ambulance is running, the streetlights will start blinking with a different color to report an emergency to nearby vehicles).
7. While nodes are connected to the virtual network, the client polls its position toward the service.
8. When the vehicle moves away from the coverage area, the service manager will remove it from the virtual network and disable the service.

Smart traffic light service (shown in Figure 17) is similar to the previous scenario with only difference that the services start and finish exploiting a timer, therefore the traffic lights will return to the initial state sequentially in order to allow the normal traffic flow. When the last traffic light turns off the green wave service, it will remove the vehicle from the virtual network.

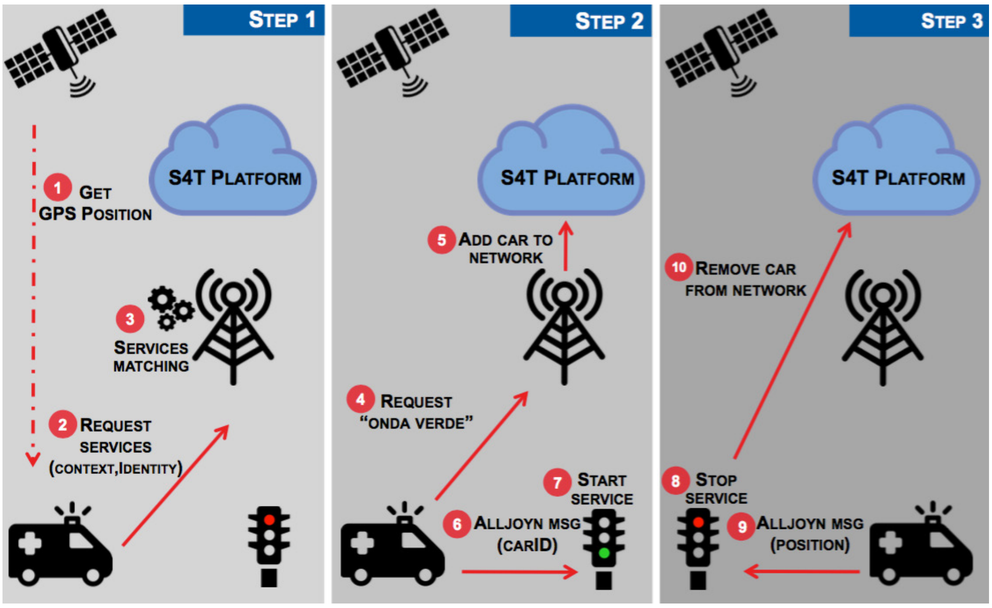

Figure 17 - The operating principle that enables the smart street traffic light service

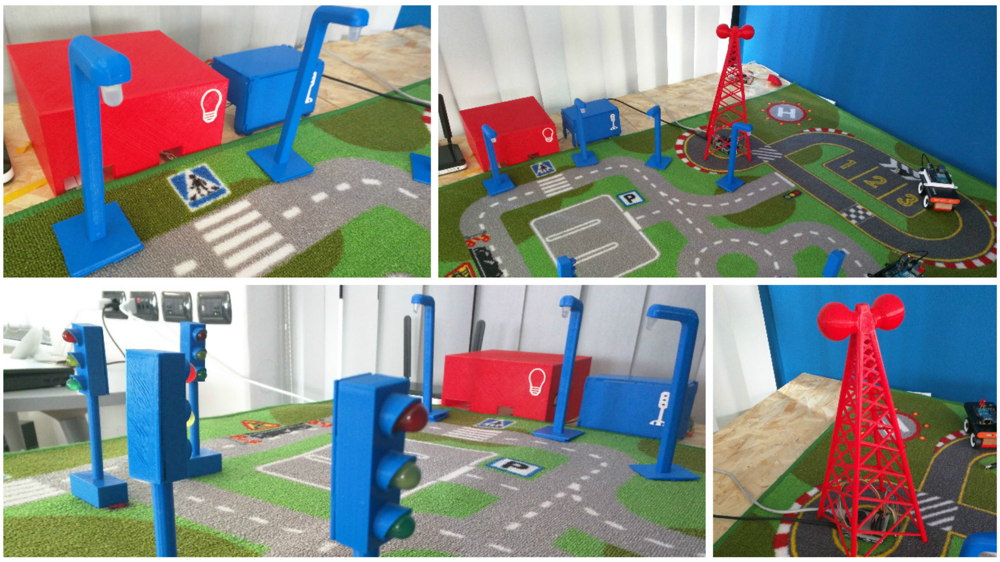

Figure 18 – Developed prototype

4.3 Proposed Architectures of MEC for critical event scenario

4.3.1 **Scenario**

Nowadays, cities are facing unprecedented challenges. The pace of urbanization is increasing exponentially, with a multitude of different kinds of information based on both infrastructure network, such as wireless sensor networks, and smart mobile objects such as smartphones and wearable.

The proposed scenario exploits the interconnection between evolved RAN, IoT and cloud, in order to analyze the information provided by several sources and detect an abnormal or critical event occurring in a smart city. The use case considers three sources:

- Personal devices, which include accelerometer, gyroscope, proximity sensor and GPS position, enabling a crowdsourcing system (CS);
- Video surveillance system (VSS), provided by the cameras network deployed in the city;
- Wireless air quality sensor system (WAQS) deployed in the city

In addition to these sources of information, users can send photos or videos acquired during a critical event. This type of information is obtained on a voluntary basis with the intent to exploit the user's curiosity and his sharing desire as a support tool for the public safety In a traditional scenario, each of the aforementioned systems sends the information to the own service provider, thus the information analysis and consequently the events notification is managed by central system, generally deployed in a cloud or host infrastructure. The disadvantages are the increasing of network traffic and the response delay, because there is no pre-data aggregation at lower levels. Moreover, the correlation of this information is not frequently possible because data belongs to different stakeholders that keep it in their private infrastructures.

In order to reduce the latency among detection and a related decision and computational effort for the processing of a request, it is possible to scale the architecture on which resides the business logic increasing the number of machines dedicated to data processing. In this approach, each machine can process a small amount of data compared to the traditional centralized case.

Additionally, moving the computation effort to closest nodes to the source of information leads to a drastic reduction of the transmission time. Moreover, in a smart city scenario, in which citizens move frequently, the concepts of mobility and position are critical to provide services or information. The technology evolution allows us to exploit these data together with innovation techniques as well as pattern recognition, data analysis and predictive analytics, enabling the distribution of computation on the smart cities with Mobile Edge Computing and Fog Computing.

4.3.2 Architecture and principle of operation of the solution

Four layers compose the architecture shown in Figure 19:

- Service on top layer, composed by services directly connected to the cloud providers;
- Cloud layer, that contains the cloud providers which exploit the information provided by a single IoT/WSN system;
- Mobile Edge Computing layer, that enables a subset of features, usually exposed at the cloud level, such as processing and analysis of data. The components deployed inside the MEC Server will be discussed below.

- Cyber-Physical Layer, composed by an heavily distributed ecosystem that cover several smart objects, mobile and fixed, such as personal devices, sensors, video surveillance systems.

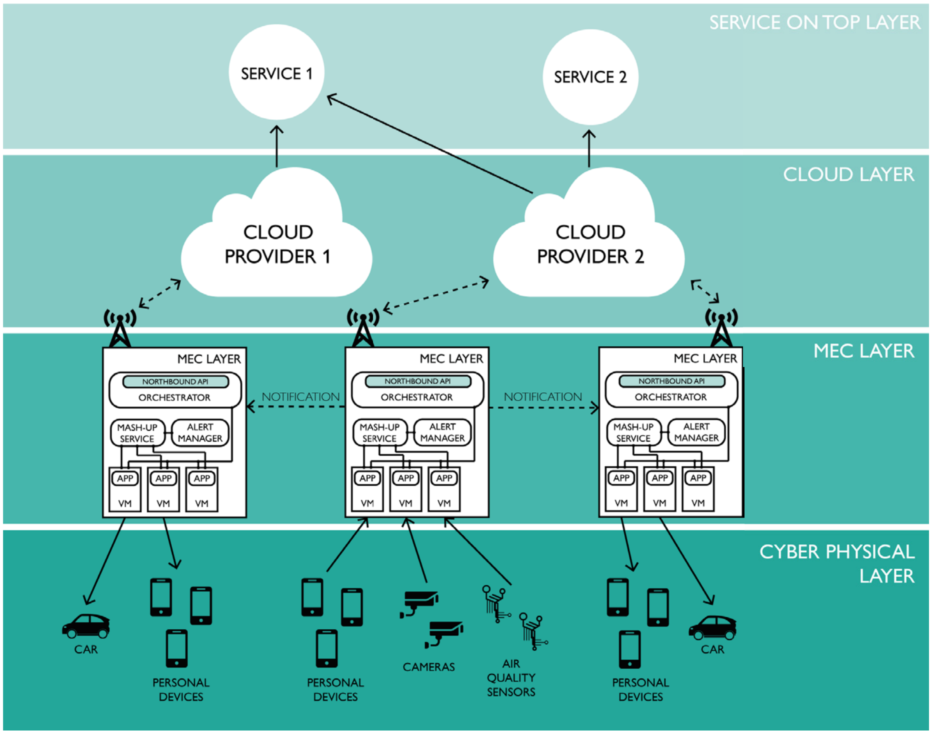

Figure 19 - Overall architecture of the proposed scenario

This solution allows the processing of every data source through the MEC Servers (shown in Figure 20) located at the MEC layer. Each self-contained application (APP), deployed in a virtual machine, manages a single source of information.

The data sources are continuously monitored by the Mash-Up Service Application, that contains the algorithms, which predict the abnormal or critical event. The algorithms analyze the traffic produced by information sources to discover a pattern that can identify a critical event. When a certain Mash Up Service detects an event, it triggers the Alert Notification Manager, which in turn sends two type of notice;

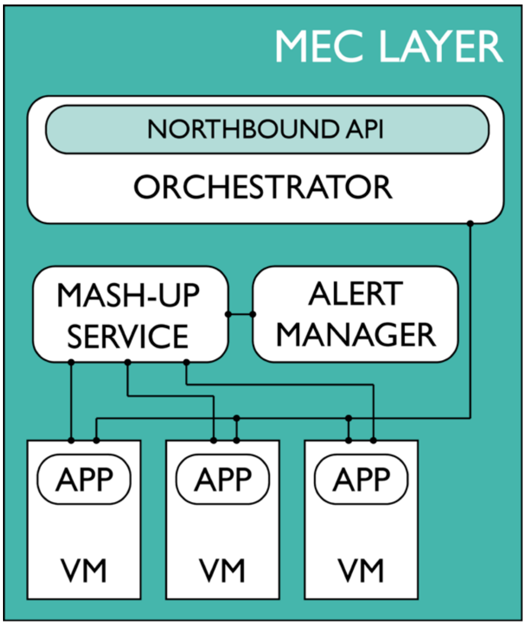

Figure 20 - MEC Server software components

the first one includes the position and the event type, to advise the neighboring eNodeBs. The second one is sent to the cloud provider, and represents a more complex information containing the type of event, the position and the data aggregation. Regarding the eNodeB neighbors, the Alert Manager deployed inside them processes the notification and sends an alert to the connected smart object, to advise the citizens that a critical event has occurred. Moreover, the MEC Server exposes the API to allow the application's provisioning by the cloud providers. Finally, an orchestrator component has been introduced to support a flexible and multi-tenancy environment and to expose the northbound API to the cloud provider that can inject plugin to modify at runtime the self contained application.

Figure 21 shows a flow diagram, which describes a specific use case when a critical event occurs.

Assuming that a fire has happened, a mass of citizens run away to refuge. The measures retrieved by motion sensors are sent to the MEC App.

The MEC App receives the measures and detects an anomaly. Thus, it sends a request to the Mash-Up service to analyze other sources and detect the specific critical event.

Through the analysis of image frames (pattern recognition) and the increase of carbon-dioxide, the event is classified as a fire alarm.

The Mash-Up service sends a request to the Alert Manager indicating the type of event and the location. The Alert Manager sends a notification to the neighboring eNodeBs, that provides the information to the connected smart objects (e.g. personal devices, car navigation systems) allowing citizens to be aware regarding the fire threat.

Furthermore, the notification, through the cloud provider, is sent to the services on top (e.g. fire fighters, police department).

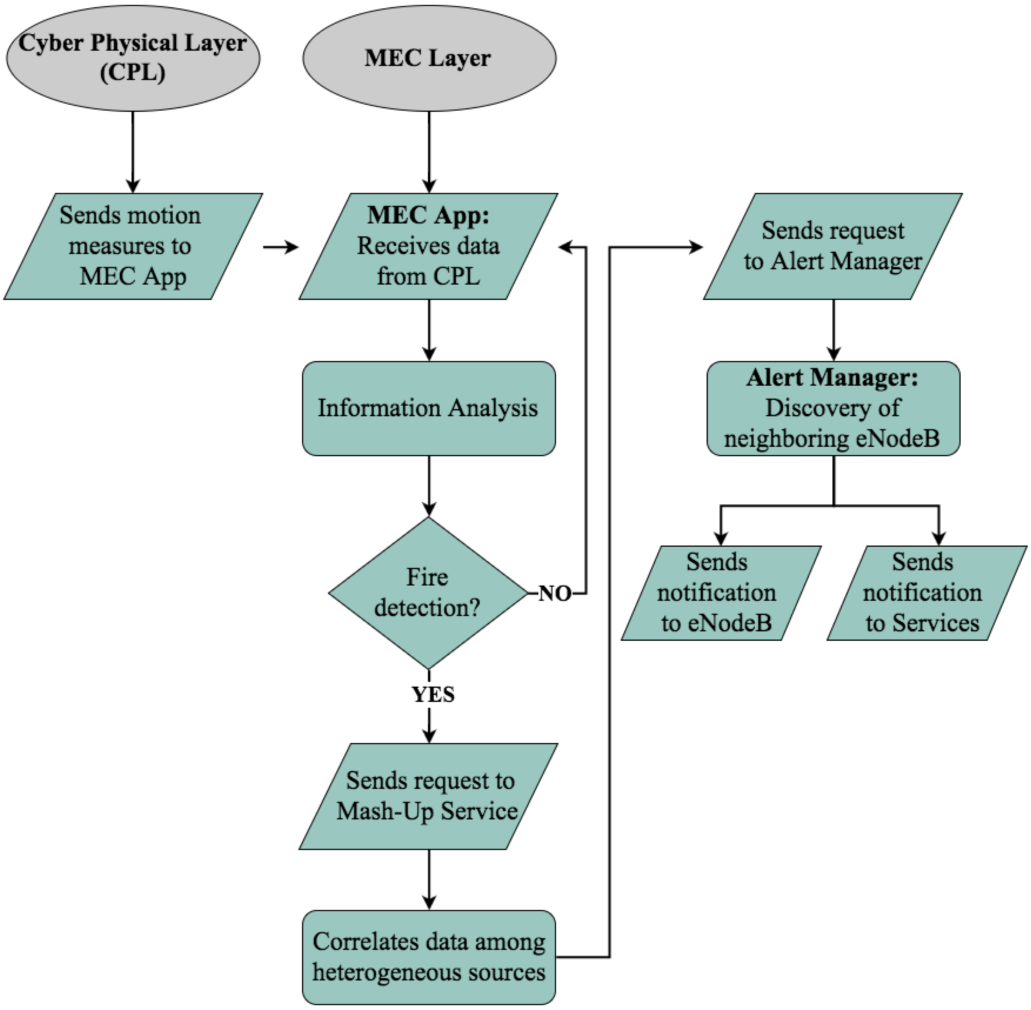

Figure 21 Flow diagram, describing the MEC logic

4.3.3 Discussion of results and possible developments

In this work, we have presented a scenario which helps to improve the services readiness required in case of time-critical events. Under these circumstances, the huge number of connected devices can affect network performances but at the same time it can contribute, thanks to user-generated contents and appropriate algorithms, to notice the occurrence of anomalous events.

Our approach distributes the computational load onto the network equipment that, in the traditional architecture, act only as a means of communication. The advent of new technologies in mobile networks allows leveraging on nodes belonging to the RAN to deploy services designed for smart city, improving the user experience of citizens. As a future work, we intend to exploit the implementation of an application inside the RAN to provide a programmable networked cloud environment that will contribute to satisfying the demanding requirements of Smart Objects and Mobile Apps in terms of expected latency, resilience, scalability.

4.4 Proposed framework for mass events monitoring through crowdsourced media analysis

4.4.1 Scenario

Nowadays, the pace of urbanization is exponentially increasing, with a multitude of information sources such as wireless sensor networks, cameras, smartphones, wearable devices and social networks. Such a huge amount and variety of data provide new technological assets to deal with unprecedented challenges in the context of smart cities.

Over the last few years, with considerable complicity of social networks, we have discovered the enormous value that people can bring to different areas by means the information they produce. Common users have become prosumer (producer-consumer) of original contents.

The scenario considered in this work regards the detection and the monitoring of abnormal events in a city with the help of its citizens. The main strength of a crowdsourced approach is the pervasiveness of the people, who can be considered as “mobile sensors”. A similar pervasiveness is hard to achieve using an infrastructure of real sensors (eg. cameras) mainly to their costs in terms of hardware, installation and management. People are instead inclined to share information and multimedia contents about their activities and the environment around them. Each user can be considered as a source of information and can contribute to improve the knowledge on the area where it is located and about a possible occurring event. The contributions from many different users can be combined to obtain an overall view of the area of interest. The proposed scenario exploits a crowdsourced approach to determine the type of event that occurs in a specific area and monitor how the situation evolves.

After an event detection notification, due to the observation of abnormal signals, the proposed framework receives the geographic information of the involved area. This information is exploited to make a request to a proper MEC application, which handles the communication between the involved users and the framework. The system empowers the BTS to broad- cast a “fast query” to the users within the interested area.

4.4.2 **Proposed Framework**

This section describes the proposed framework in order to achieve a quick and efficient monitoring and being also able to identify the specific critical event, warn citizens and call the rescue services.

Initially, the event detection is carried out evaluating the sudden movement of a large mass of people, through the sensors on the personal devices, such as: accelerometer, gyro- scope, proximity sensor and GPS position, enabling an implicit crowdsourcing.

The main components of the framework, shown in Figure 22, are the following:

- Service Interface: this component provides the access API to the framework's capabilities, which can be properly exploited by an external service;
- Query Engine: this module is devoted to the definition of fast queries used to interact with users, in a simple and immediate way, with the aim of acquiring general information about the status of a specific area affected by an event. Moreover this module is also responsible for generating multimedia contribution request to improve the awareness about the area of interest.
- Crowdsourced Monitor: this module includes a set of algorithms devoted to perform analysis and the inference on the user gathered contents. The obtained results are provided to the services on TOP and affect the user profiling.

- User Profiling: this component collects historical data about users interactions and performs a profiling based on the quality of the obtained information and the used devices.
- RAN Interface: this module allows the interaction between the proposed framework and the RAN by means a proper API.

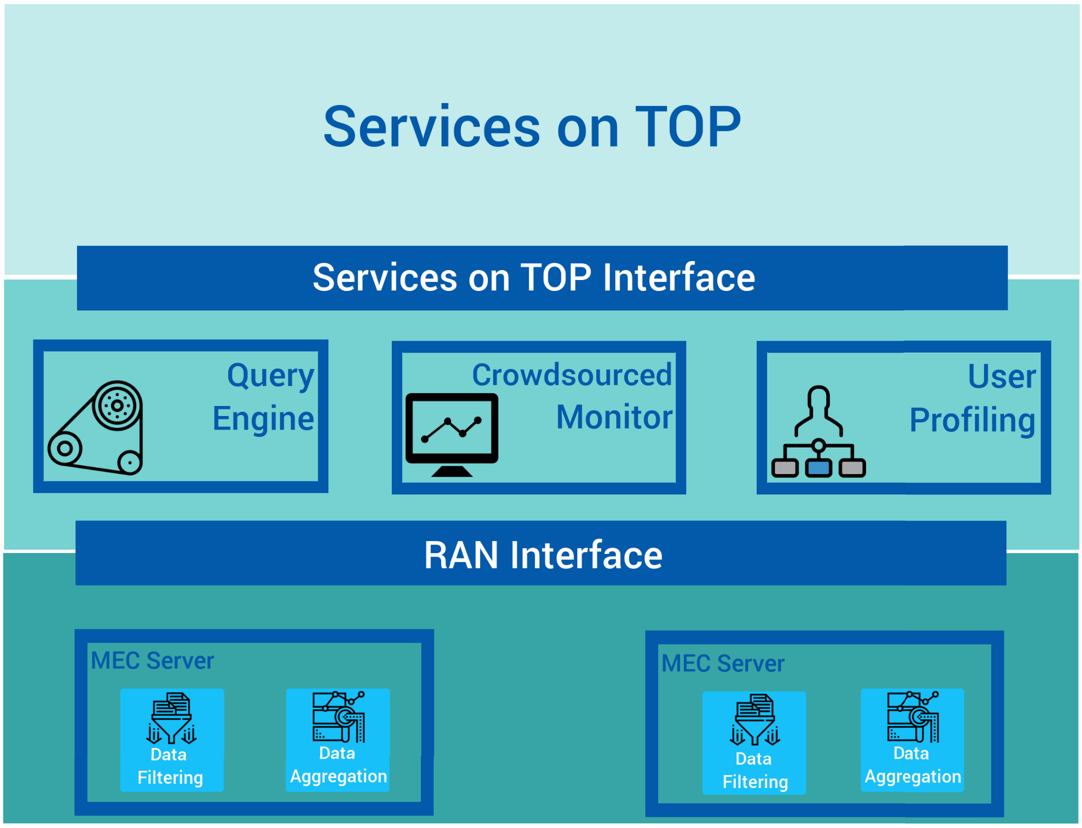

Figure 22 – Proposed framework for Mass Event Monitoring

In Figure 23, the details of the monitoring process are shown. The monitoring process may be triggered by an automatic system of event detection or by an operator to gain awareness of what is happening in the affected area and acquire additional details about the event in progress. The numbered arrows describe a typical execution flow. Here follows a step-by-step description of the actions taken by the system to serve the request:

1) The framework receives through the *Service On Top Interface* a monitoring request for a specific area.

2) The *Query Engine* processes the received request and prepare a set of fast queries to send to the users located in the area of interest.

3) The *RAN Interface* forwards the query messages to the specific BTS, which covers the area and sends broadcast text messages, containing the short questions. 4) The MEC Application deployed on the *MEC Server* analyzes the users' answers and filters them according to the information accuracy (e.g., event awareness).

5) The *Query Engine* combines the acquired information with the historical users' profile data (e.g., trusted users, device model).

6) The *Query Engine* exploits this information to define the suitable contribution request.

7) The *RAN Interface* sends the contribution request to the specific BTS which forwards it to the users demanding multimedia contribution (e.g., video stream, pictures).

8) In this step, the MEC Application performs some pre- processing on the multimedia contributes. For instance, if the number of provided video streams is high, they can be filtered considering several quality factors (e.g., video resolution, frame rate, stability). This prevents the system to elaborate noisy information, and the reduction of the amount of the data to process.

9) The *Crowdsourcing Monitor* performs the analysis of the acquired data. These analysis depend on the kind of the required data and the aim of the analysis. For instance, if the system requires to perform the visual monitoring of the area of interest, the video streams provided by the users can be clustered according to the visual content with the aim to understand what is the most viewed scene [8].

The whole above described process is made transparent to the user by means of both the mentioned interfaces. The system only requires the area of interest to be monitored.

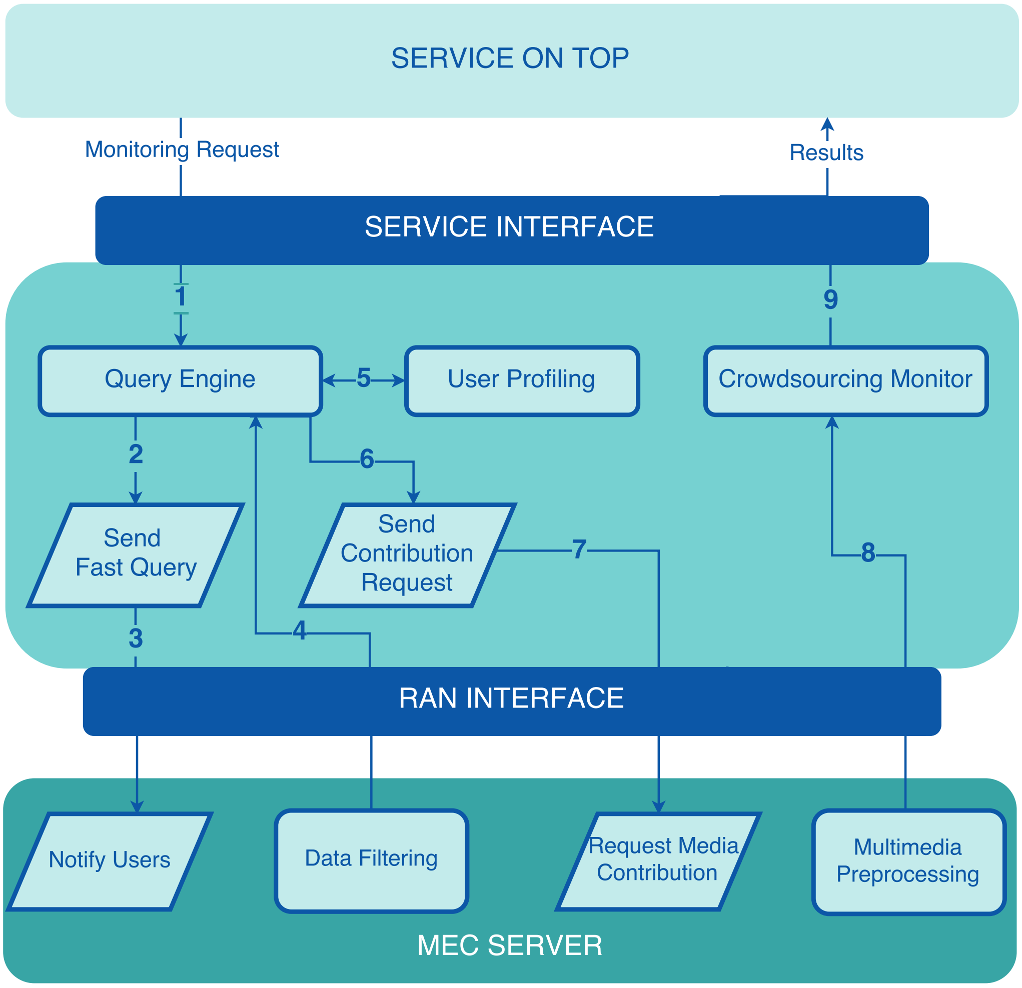

Figure 23 - A flow diagram that describes the step-by-step interactions among the different modules of the proposed framework

4.4.3 Discussion of results and possible developments

This work presents a framework able to provide a quickly monitoring of critical events, obtained by exploiting a crowdsourcing approach and Mobile Edge Computing paradigm. Upon receiving a monitoring request, the system defines a process that allows to collect, analyze and make proper inferences on crowdsourced data, with the aim to provide useful results to the service.

The contributes of the users are selected considering several factors including user proximity, quality of the provided information, user profiling, as well as the user answer to a fast query. The collected data are then aggregated and analyzed. Examples of contributes aggregation analysis are visual clustering and saliency of video streams. The analysis results are then provided to the Service On Top and exploited to update the users' profiles.

In the next steps of this project, we are building up an actual prototype of the proposed framework with the aim to achieve experimental results and assess the effectiveness of the method described in this paper.

In the last few years, emerging technologies have been leading toward the realization of efficient cities, where technology will be applied to improve and support citizens' daily living in a realistic Smart City scenario in which the proposed framework is fully integrated.

5 Conclusion

In this thesis we discussed several issues related to Internet of Things and the rapid growth of the connected devices, proposing different solutions to these challenges spanning both theoretical aspects and practical implications exploiting the Edge Computing paradigm.

The aim of these works was to demonstrate how Fog Computing and Mobile Edge Computing can bring significant benefits to IoT such as smart cities and smart agriculture scenarios.

The proposed framework related to precision agriculture, exploiting a 2-tier Fog architecture allows to greatly reduce the transmitted and stored data to the Cloud.

With regard to smart city scenarios, the work carried out concerned both smart traffic light and smart street light services, as well as monitoring and managing critical events using the Mobile Edge Computing paradigm.

In both cases the aim was to distribute the the computational load onto the network equipment that, in the traditional architecture, act only as a means of communication.

In this way, time critical services are more efficient, greatly improving the quality of life of citizens.

Edge computing is ideal in contexts where data transmitted to the cloud for analysis and processing would negatively affect performance and where connectivity is intermittent, such as rural areas. It supports multiple vertical sectors and applications enabling systems to become more flexible, economically advantageous, safe and scalable.

Therefore, all that demonstrates how Edge Computing is emerging as the best choice to bridge the gap between the IoT and the cloud.

The work presented in this thesis have appeared in the articles below or presented in important scientific events.

List of Publications

Journal papers

Ermanno Guardo, Alessandro Di Stefano, Aurelio La Corte, Marco Sapienza, Marialisa Scatà. **A Fog Computing-based IoT Framework for Precision Agriculture.** Journal of Internet Technology (to appear, January 2019)

Scatà, M., Di Stefano, A., La Corte, A., Liò, P., Catania, E., Guardo, E., & Pagano, S. **Combining evolutionary game theory and network theory to analyze human cooperation patterns.** (2016) *Chaos, Solitons & Fractals*, *91*, 17-24.

Di Stefano, A., Scatà, M., La Corte, A., Liò, P., Catania, E., Guardo, E., & Pagano, S. **Quantifying the role of homophily in human cooperation using multiplex evolutionary game theory**. (2015) *PloS one*, *10*(10), e0140646.

Emanuele Catania, Alessandro Di Stefano, Ermanno Guardo, Aurelio La Corte, Salvatore Pagano, Marialisa Scatà. **Energy Awareness and the Role of "Critical Mass" In Smart Cities.** International Refereed Journal of Engineering and Science (IRJES), Volume 4, Issue 7, July 2015, pp. 38-43. ISSN: 2319-183X (online); 22319-1821 (print)

Conference papers

Marco Cavallo, Ermanno Guardo, Giuseppe La Torre Alessandro Ortis, Marco Sapienza. **Mass Events Monitoring Through Crowdsourced Media Analysis** (2017) PATTERNS, The Ninth International Conferences on Pervasive Patterns and Applications, IARIA

Sapienza, M., Guardo, E., Cavallo, M., La Torre, G., Leombruno, G., & Tomarchio, O. **Solving critical events through mobile edge computing:**

An approach for smart cities. (2016, May). In *Smart Computing (SMARTCOMP), 2016 IEEE International Conference on* (pp. 1-5). IEEE.

Pietro Motta, Antonio Castiglione, Emanuele Catania, Ermanno Guardo, Aurelio La Corte, Salvatore Pagano. **Benefits of Nanosatellite Network for Smart Metering Technological Infrastructure in Wide Areas.** Tartu Conference on Space and Technology, Tartu, Estonia, September 22-24, 2014, poster & invited talk

Guardo, E., Morabito, G., Catania, G., Mursia, A., & Battiati, F. (2014, May). **BRAVO: A black-hole resilient ad-hoc on demand distance Vector rOuting for tactical communications**. In Communications and Networking (BlackSeaCom), 2014 IEEE International Black Sea Conference on (pp. 127-128). IEEE.

Bibliography

[1] Parkhill, D. F. Challenge of the computer utility. (1966).

[2] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., ... & Zaharia, M. A view of Cloud Computing. Communications of the ACM, 53(4), 50-58. (2010).

[3] Chang, K. D., Chen, C. Y., Chen, J. L., & Chao, H. C. Internet of things and cloud computing for future internet. In *Security-Enriched Urban Computing and Smart Grid.* Springer Berlin Heidelberg. (pp. 1-10). (2011).

[4] Atzori, L., Iera, A., Morabito, G.: The internet of things: a survey. Comput. Netw. 54(15), 2787–2805 (2010)

[5] Alba, E.: Intelligent systems for smart cities. In: Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary Computation Conference, pp. 707–722. ACM (2015)

[6] Di Stefano, A., La Corte, A., Leotta, M., Lió, P., & Scatá, M. It measures like me: An IoTs algorithm in WSNs based on heuristics behavior and clustering methods. Ad Hoc Networks, 2637-2647, 11(8), (2013).

[7] Stankovic, J. Research directions for the internet of things. Internet of Things Journal, IEEE, 1(1), 3-9. (2014).

[8] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 1645-166029 (7). (2013).

[9] Jin, J., Gubbi, J., Marusic, S., & Palaniswami, M. An information framework for creating a smart city through Internet of things. Internet of Things Journal, IEEE, 1(2), 112-121, (2014).

[10] Castro, M., Jara, A. J., & Skarmeta, A. F. Smart lighting solutions for smart cities. In Advanced Information Networking and Applications Workshops (WAINA), 27th International Conference on IEEE (pp. 1374-1379) (March, 2013).

[11] Dey, S., Chakraborty, A., Naskar, S., & Misra, P. Smart city surveillance: Leveraging benefits of cloud data stores. In Local Computer Networks Workshops (LCN Workshops), IEEE 37th Conference on IEEE (pp. 868-876), October 2012.

[12] Chin-Feng Lai, Sherali Zeadally, Jian Shen, Ying-Xun Lai, "A Cloud-Integrated Appliance Recognition Approach over Internet of Things," Journal of Internet Technology, Vol. 16 No. 7, PP. 1157-1168, 12, 2015.

[13] Yan-e, D. (2011, March). Design of intelligent agriculture management information system based on IoT. In Intelligent Computation Technology and Automation (ICICTA), International Conference on IEEE (Vol. 1, pp. 1045-1049). 2011.

[14] Yujun Ma,, Yin Zhang, Ong Mau Dung, Ran Li, Daqiang Zhang, "Health Internet of Things: Recent Applications and Outlook," Journal of Internet Technology, Vol. 16 No. 2, PP. 351-362, March 2015.

[15] Chung-Hsin Liu, Le Le Trung, "The Study of Using Big Data to Solve Medical Volunteer Problem," Journal of Internet Technology, Vol. 16 No. 3, PP. 416-431, 5 2015.

[16] Ganz, F., Barnaghi, P., Carrez, F., & Moessner, K. Context-aware management for sensor networks. In Proceedings of the 5th International Conference on Communication System Software and Middleware ACM (p. 6) (2011, July).

[17] Cisco, C. V. N. I. Global Mobile Data Traffic Forecast Update. 2014–2019 (white paper). (2015).

[18] Cisco, I. (Cisco visual networking index: Forecast and methodology, 2014–2019 (white paper), (2015).

[19] Di Stefano, A., La Corte, A., Lió, P., & Scatá, M. Bio-Inspired ICT for Big Data Management in Healthcare. In Intelligent Agents in Data-intensive Computing. Springer International Publishing. (pp. 1-26), (2016).

[20] Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. Fog computing and its role in the internet of things. In Proceedings of the first edition of the MCC workshop on Mobile cloud computing (pp. 13-16). ACM. August, 2012.

[21] Bonomi, F., Milito, R., Natarajan, P., & Zhu, J. Fog computing: A platform for internet of things and analytics. In Big Data and Internet of Things: A Roadmap for Smart Environments Springer International Publishing, (pp. 169-186), (2014).

[22] Patel, M., Naughton, B., Chan, C., Sprecher, N., Abeta, S., &Neal, A. (2014). Mobile Edge Computing Introductory Technical White Paper. *White Paper, Mobile-edge Computing (MEC) industry initiative*

[23] Laurence Sheriff. Back to the future 2015, https://www.planning.org.nz/Attachment?Action=Download.

[24] AliveCor Partners with LifeWatch to Expand Into Remote Patient Monitoring https://www.alivecor.com/press/press_release/alivecor-partners-with-lifewatch-to-expand-into-remote-patient-monitoring/, AliveCor. 2016.

[25] Artbrew is an automated beer brewing machine. http://blog.atmel.com/2016/03/30/artbrew-is-an-automated-beer-brewing-machine/, Atmel, 2016.

[26] Ron Miller. Cisco to buy Jasper technologies for $1.4 billion. Techcrunch 2016

[27] Big Data in IoT: Key Trends, Opportunities and Market forecasts 2015-2020. Mind Commerce Report 2015

[28] How IoT is Empowering Transportation Industry. http://www.cxotoday.com/story/how-iot-is-empowering-tl-industry/, CXO today news desk, 2015

[29] IEEE Standard for Information Technology- Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003), pp. 0 1–305, 2006. [Online]. Available: http://dx.doi.org/10.1109/ieeestd.2006.232110

[30] B. P. Crow, I. Widjaja, L. Kim, and P. T. Sakai, "Ieee 802.11 wireless local area networks," Communications Magazine, IEEE, vol. 35, no. 9, pp. 116–126, 1997

[31] S. Bluetooth, "Bluetooth core specification v4.0," 2010

[32] S. Tozlu, "Feasibility of wi-fi enabled sensors for internet of things," in Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International. IEEE, 2011, pp. 291–296

[33] C. Gomez, J. Oller, and J. Paradells, "Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology," Sensors, vol. 12, no. 9, pp. 11 734–11 753, 2012

[34] J. Postel, "Rfc 791: Internet protocol," 1981

[35] Rfc 793: Transmission control protocol," 1981

[36] J. Postel, "Rfc 768: User datagram protocol," 1980

[37] S. R. Das, E. M. Belding-Royer, and C. E. Perkins, “Ad hoc on- demand distance vector (aodv) routing,” 2003

[38] “MQTT,” April 2012. [Online]. Available: http://mqtt.org

[39] Z. Shelby, K. Hartke, and C. Bormann, “Constrained application protocol (coap),” 2013

[40] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, “Hypertext transfer protocol–http/1.1,” 1999

[41] R. T. Fielding and R. N. Taylor, “Principled design of the modern web architecture,” ACM Trans. Internet Technol., pp. 115–150, May 2002

[42] J. Dilley et al., “Globally Distributed Content Delivery,” *IEEE Internet Computing*, vol. 6, no. 5, 2002, pp. 50–58.

[43] IBM Corporation, “Smarter wireless networks; add intelligence to the mobile network edge.”

[44] Intel and Nokia Siemens Networks, “Increasing mobile operators’ value proposition with edge computing.”

[45] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, “A survey of mobile cloud computing: architecture, applications, and approaches: A survey of mobile cloud computing,” Wireless Communications and Mobile Computing, vol. 13, pp. 1587–1611, Dec. 2013

[46] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, “The case for vm-based cloudlets in mobile computing,” Pervasive Computing, IEEE, vol. 8, no. 4, pp. 14–23, 2009

[47] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing.

[48] Fernando, N., Loke, S. W., & Rahayu, W. (2013). Mobile cloud computing: A survey. Future Generation Computer Systems, 29(1), 84-106.

[49] Yi Xu, Jun Yan, "A Cloud-Based Design of Smart Car Information Services," Journal of Internet Technology, Vol. 13 No. 2, PP. 317-326, March 2012.

[50] Vaquero, L. M., & Rodero-Merino, L. (2014). Finding your way in the fog: Towards a comprehensive definition of fog computing. ACM SIGCOMM Computer Communication

[51] an open-source LISP implementation for Linux, Android and Open- WRT. http://lispmob.org/, 2015.

[52] F. Bonomi, "Connected vehicles, the internet of things, and fog computing," in The Eighth ACM International Workshop on Vehicular Inter- Networking (VANET), Las Vegas, USA, 2011

[53] Sheng Wen," The Fog Computing Paradigm: Scenarios and Security Issues", School of Information Technology, Deakin University, 220 Burwood Highway, Burwood, VIC, 3125, Australia.

[54] Tom H. Luan, Longxiang Gao, Yang Xiang," Fog Computing: Focusing on Mobile Users at the Edge", School of Information Technology Deakin University, Burwood Melbourne, VIC 3125, Australia.

[55] Ivan Stojmenovic, "The fog computing Paradigm: Scanerios And Security Issues", SIT, Deakin University, Burwood, Australia and SEECS, University of Ottawa, Canada.

[56] Patel, M., Naughton, B., Chan, C., Sprecher, N., Abeta, S., & Neal, A. (2014). Mobile-Edge Computing Introductory Technical White Paper. White Paper, Mobile-edge Computing (MEC) industry initiative.

[57] Ahmed, A., & Ahmed, E. (2016, January). A survey on mobile edge computing. In *Intelligent Systems and Control (ISCO), 2016 10th International Conference on* (pp. 1-8). IEEE.

[58] Godfray, H. C. J., Beddington, J. R., Crute, I. R., Haddad, L., Lawrence, D., Muir, J. F., ... & Toulmin, C. (2010). Food security: the challenge of feeding 9 billion people. science, 327(5967), 812-818.

[59] Tilman, D., Balzer, C., Hill, J., & Befort, B. L. (2011). Global food demand and the sustainable intensification of agriculture. Proceedings of the National Academy of Sciences, 108(50), 20260-20264.

[60] Pierce, F. J., & Nowak, P. (1999). Aspects of precision agriculture. Advances in agronomy, 67, 1-85.

[61] Robert, P. C. (2002). Precision agriculture: a challenge for crop nutrition management. Plant and soil, 247(1), 143-149.

[62] Firdhous, M., Ghazali, O., & Hassan, S. (2014). Fog computing: Will it be the future of cloud computing? The Third International Conference on Informatics & Applications (ICIA2014).

[63] LoRa" Product Family, available at http://www.semtech.com/wireless-rf/lora.html.

[64] LoRa Alliance, “LoRa Specification V1.0,” Tech. Rep., May 2015.

[65] Centenaro, M., Vangelista, L., Zanella, A., & Zorzi, M. (2015). Long-Range Communications in Unlicensed Bands: the Rising Stars in the IoT and Smart City Scenarios. arXiv preprint arXiv:1510.00620.

[66] Stafford, J. V. (2000). Implementing precision agriculture in the 21st century. Journal of Agricultural Engineering Research, 76(3), 267-275.

[67] Bouma, J. (1997, January). Precision agriculture: introduction to the spatial and temporal variability of environmental quality. In Ciba Foundation Symposium 210-Precision Agriculture: Spatial and Temporal Variability of Environmental Quality (pp. 5-17). John Wiley & Sons, Ltd.

[68] Lampkin, V., Leong, W. T., Olivera, L., Rawat, S., Subrahmanyam, N., Xiang, R., ... & Locke, D. (2012). Building smarter planet solutions with mqtt and ibm websphere mq telemetry. IBM Redboo.

[69] Luzuriaga, J. E., Perez, M., Boronat, P., Cano, J. C., Calafate, C., & Manzoni, P. (2015, January). A comparative evaluation of AMQP and MQTT protocols over unstable and mobile networks. In 2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC) (pp. 931-936). IEEE.

[70] Collina, M., Bartolucci, M., Vanelli-Coralli, A., & Corazza, G. E. (2014, September). Internet of Things application layer protocol analysis over error and delay prone links. In 2014 7th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop.

[71] Ruß, G., Schneider, M., & Kruse, R. (2010). Hierarchical Spatial Clustering for Management Zone Delineation in Precision Agriculture. In Industrial Conference on Data Mining-Workshops (pp. 95-104).

[72] Thakare, Vishakha S., et al. "Design of smart traffic light controller using embedded system." *ISOR-JE* 10.1 (2013): 30-3.

[73] A Novel Approach to Implement Green Wave system and Detection of Stolen Vehicles.

[74] Alliance, A. (2016). Alljoyn framework. *Linux Foundation Collaborative Projects. URl: https://allseenalliance. org/framework (visited on 09/14/2016)*.

[75] http://nodejs.org/

[76] Bruneo, Dario, et al. "Stack4Things as a fog computing platform for Smart City applications." *Computer Communications Workshops (INFOCOM WKSHPS), 2016 IEEE Conference on*. IEEE, 2016.