Title:

Hierarchical Distributed Fog-to-Cloud Data Management in Smart Cities

Thesis presented in Fulfilment of the requirements for the degree of Doctor for the UniversitatPolitecnica de Catalunya
Research Group: CRAAX

By:

Amir Sinaeepourfard

Advisor:

Dr. Jordi Garcia

Co-Advisor:

Dr. Xavi Masip

Universitat Politecnica de Catalunya
Departament d'Arquitectura de Computadors
September, 2017

Acknowledgment:

Special thanks to my family. Words cannot express how grateful I am to my mother, my father, and my two brothers for all of the sacrifices that you've made on my behalf. Your prayer for me was what sustained me thus far. I would also like to thank all of my friends who supported me in writing and incented me to strive towards my goal.

I would like to express my special appreciation and thanks to my advisor and co-advisor, you have been a tremendous mentor for me. I would like to thank you for encouraging my research and for allowing me to grow as a research scientist. Your advice on both research as well as on my career have been priceless. I would also like to thank my committee members for serving as my committee members even at hardship. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you. I would especially like to thank my colleagues in CRAAX Laboratory.

Barcelona, September 2017

Amir Sinaeepourfard

Table of contents:

List of Figures:

List of Tables:

List of Abbreviations:

Number	Full name	Abbreviation
1	Data LifeCycle	DLC
2	Smart City Data LifeCycle	SCC-DLC
3	Comprehensive Agnostic Data LifeCycle	COSA-DLC
4	F2C	Fog-to-Cloud
5	Internet of Things	IoT
6	Radio Frequency Identification	RFID
7	Wireless Sensors Network	WSN
8	Quality-of-Information	QoI
9	Quality Control	QC
10	Quality Assurance	QA
11	Extraction Transformation Loading	ETL
12	Global Sensor Networks	GSN
13	Sensor Metadata Repository	SMR
14	Content Delivery Network	CDN
15	Application Programming Interface	API
16	Distributed File System	DFS
17	InterPlanetary File System	IPFS
18	Distributed Hash Table	DHT

Abstract

There is a vast amount of data being generated every day in the world, coming from a variety of sources, with different formats, quality levels, etc. This new data, together with the archived historical data, constitute the seed for future knowledge discovery and value generation in several fields of science and big data environments. Discovering value from data is a complex computing process where data is the key resource, not only during its processing, but also during its entire life cycle. However, there is still a huge concern about how to organize and manage this data in all fields, and at all scales, for efficient usage and exploitation during all data life cycles. Although several specific Data LifeCycle (DLC) models have been recently defined for particular scenarios, we argue that there is no global and comprehensive DLC framework to be widely used in different fields.

In particular scenario, smart cities are the current technological solutions to handle the challenges and complexity of the growing urban density. Traditionally, Smart City resources management rely on cloud based solutions where sensors data are collected to provide a centralized and rich set of open data. The advantages of cloud-based frameworks are their ubiquity, as well as an (almost) unlimited resources capacity. However, accessing data from the cloud implies large network traffic, high latencies usually not appropriate for real-time or critical solutions, as well as higher security risks. Alternatively, fog computing emerges as a promising technology to absorb these inconveniences. It proposes the use of devices at the edge to provide closer computing facilities and, therefore, reducing network traffic, reducing latencies drastically while improving security. We have defined a new framework for data management in the context of a Smart City through a global fog to cloud resources management architecture. This model has the advantages of both, fog and cloud technologies, as it allows reduced latencies for critical applications while being able to use the high computing capabilities of cloud technology.

In this thesis, we propose many novel ideas in the design of a novel F2C Data Management architecture for smart cities as following. First, we draw and describe a comprehensive scenario agnostic Data LifeCycle (COSA-DLC) model successfully addressing all challenges included in the 6Vs, namely Value, Volume, Variety, Velocity, Variability and Veracity, not tailored to any specific environment, but easy to be adapted to fit the requirements of any particular field. Then, we introduce the Smart City Comprehensive Data LifeCycle (SCC-DLC) model, a data management architecture generated from a comprehensive scenario agnostic model, tailored for the particular scenario of Smart Cities. We define the management of each data life phase, and explain its implementation on a Smart City with Fog-to-Cloud (F2C) resources management. And then, we illustrate a novel architecture for data management in the context of a Smart City through a global fog to cloud resources management architecture. We show this model has the advantages of both, fog and cloud technologies, as it allows reduced latencies for critical applications while being able to use the high computing capabilities of cloud technology. As a first experiment for the F2C data management architecture, a real Smart City is analyzed, corresponding to the city of Barcelona, with special emphasis on the layers responsible for collecting the data generated by the deployed sensors. The amount of daily sensors data transmitted through the network has been estimated and a rough projection has been made assuming an exhaustive deployment that fully covers all city. And, we provide some solutions to both reduce the data transmission and improve the data management. Then, we used some data filtering techniques (including data aggregation and data compression) to estimate the network traffic in this model during data collection and compare it with a traditional real system. Indeed, we estimate the total data storage sizes through F2C scenario for Barcelona smart cities.

Keywords — Data LifeCycle, Data Management, Fog-to-Cloud (F2C), Vs Challenges

Chapter 1:

Introduction

1.1 Motivation

It is expected that 70% of the world's population will live in cities and surrounding areas by 2050. Municipal managers have to devise new ways to managing and organizing the city in order to mitigate the issues derived from such amount of population, while maintaining or even increasing the citizens' quality of life. Smart cities are the technological solutions designed, not only for absorbing the increasing pressure of population, but mainly for supplying better and more efficient services and processes, promoting a sustainable economic growth and, consequently, providing a higher quality of life to citizens [1, 2].

Smart cities involve different challenging technologies, and demand an exhaustive deployment of computing resources throughout the city (from sensors networks or mobile smart devices, to powerful data centers), all connected through several communication networks using different technologies (wireless sensor networks, 4G, WiFi, Bluetooth, etc.), and all together managed and coordinated by deploying sophisticated frameworks. However, beyond all foreseen but also unforeseen technologies, the most precious resource for a city to become smart is data.

A huge amount of data is constantly being produced in the world, turning Big Data into one of the hottest research topics currently. Data are being generated from multiple scientific sources, including Smart Cities, the IoT, scientific modeling, or big data simulations [3-5]; but also from users' social, professional or everyday activities. These daily fresh data are accumulated over other historical repositories, setting up the complex universe of digital data. This data can then be used in different forms during big data processing (reading, writing, transforming or removing), and then be reused in further processes, therefore drawing the life cycle of data.

Data are the fuel for the Smart Cities technology. Indeed, data allow a city to become smart, instead of just automatized. This is rooted to the fact that data provide the required information for services to proceed according to contextual parameters, or some higher value knowledge extracted from complex data analysis. In fact, Smart Cities constitute the ideal scenario to generate abundant data from any kind of source, such as the own city's sensors, participatory sensing (for instance, sensors integrated in citizens' smartphones), data obtained from social media or any other third party application, surveillance cameras and devices, or any other city resource sensitive to contribute with additional information. For this reason, many efforts from academia and industry are being devoted to create and use data analysis algorithms in order to take advantage of this tremendous abundance of data. However, not many researchers are paying attention to explicit and efficient data management strategies in the context of Smart Cities.

Data management and organization during their entire life cycle, including data generation, data acquisition, data preservation, or data processing, becomes a complex and challenging task [6, 7]. The main objective of data management is to provide easy and safe access to data sources and repositories, in order to be able to extract any form of value through complex computing and analytical processes aver big data sources. For this reason, efficient data management and organization systems are a key topic for effective value generation. In addition, the traditional concepts of Relational Database Management Systems (RDBMS) and the recent Extract-

Transform-Load (ETL) process have been proposed for modeling the typical data life cycles in data warehousing environments [5, 8, 9]. Moreover, it is worth highlighting that Big Data brings further challenges to the traditional data management and organization systems [5, 10].

Several Data LifeCycle (DLC) models have been proposed as an effective data management solution that facilitates the extraction of knowledge in complex data systems (see for example references [6, 7, 11-13]). In a short, a DLC models define the sequence of phases in the data life, specify the management policies for each phase, and describe the relationship among phases [11]. Furthermore, a DLC model is specific for a particular field and scenario, addressing their private requirements and challenges [7, 13]. Thus, the main goal of a Data LifeCycle model is to optimize data management – e.g., considering an efficient organization or the removal of any kind of waste– in order to offer end-users data products best suiting the expected quality requirements [11, 14, 15].

1.2 Problem Statement

In this thesis, we aim at contributing to the problem of designing a comprehensive data management architecture in the context of a Smart City. We believe data are the most important resource in a Smart City; without an efficient management of the data obtained from the whole deployed sensors network, no smart services can be provided. Such a problem faces with the following difficulties:

- Data management in the context of smart cities is a well-known complex topic, there is no contribution yet considering data management as a nuclear component in the design of a Smart City architecture.
- There is significant and unstoppable activity in the scientific community turning into new architectures and models for managing computation at the edge (understanding this paradigm either as edge computing or fog computing), but none of the proposed solutions considers the data as the central management resource.
- There is not any data management architecture designed integrally throughout all data life cycle stages, from collection to removal, including storage and processing.
- As of today, there is no data management architecture designed comprehensively, intended to address the main challenges of the Big Data paradigm applied to a Smart City scenario, which can be summarized as the 6Vs challenges, which are:
    - VOLUME: In a Smart City there is a wide sensors network deployed throughout the whole city. Such network is undoubtedly generating huge volumes of data that must be efficiently managed to make them accessible for users and applications.
    - VELOCITY: The data generation speed rate can be really high, and an effective data management system has to be able to deal with such velocity.

  - VARIETY: The data types and formats generated are heterogeneous, since different types of sensors generate different types of data, using distinct formats, templates and characteristics.
  - VARIABILITY: In a Smart City, each data source continuously generates new updated versions of the produced data. The updated version may overwrite the last version when according to the specific data updating policy in place.
  - VERACITY: The exponential data growth is unquestionably posing tremendous constraints and demands in the Smart City, such as data quality or data security.
  - VALUE: And the last, but actually the most important challenge, is value: the whole data management architecture has to be defined to provide a framework to use data and generate information, knowledge, intelligence and, in summary, smartness to the city services.

- Data sources in a Smart City are widely distributed. Such as a distributed topology drives complex challenges for designing an effective data management architecture such as providing low latencies rates, reducing data transfer volumes, or providing efficient performance for both, critical services but also high performance computing applications. All these requirements have to be provided globally.

1.3 Research Objectives

In this thesis we aim at designing a data management architecture for smart cities. We envision a distributed architecture with a hierarchical structure, meeting the following objectives:

- The architecture must be distributed and hierarchical, effectively defining and coordinating all elements that belong to the system to become a broad, integrated, and efficient system.
- The architecture must be comprehensive, i.e., it must have clear design and definitions for all data life cycle stages, including data acquisition, data processing and data preservation.
- The architecture must address the 6Vs challenges as part of its design, becoming efficient, flexible, and providing high data quality. For this reason, the architecture must provide:
  - Efficiency for managing high volumes of data and at high speed rate.
  - Flexibility for managing heterogeneous data types and formats.
  - Quality, providing clean and polished data and, consequently, contributing to improve the system efficiency as well.
- The architecture must be able to provide fast data access to match efficiently the requirements of both real-time and critical applications, by exploiting the features of data sources locality and fog computing.

- The architecture must be able to provide efficient data access to historical (not recent) data repositories to match the requirements of deep computing and big data applications, by exploiting the capabilities of cloud computing systems.
- The architecture must be able to cover all sections and districts of the city and, therefore, become fully scalable without losing efficiency.

Aligned to the proposed research objectives, in this thesis we propose a new architecture for data management in the context of smart cities. We base our architecture on the definition of a comprehensive scenario agnostic DLC model, named the SCC-DLC model. This model has been proved to address the 6Vs data challenges, and has already been easily adapted to the specific scenario of a Smart City. As a reference contribution for our proposed architecture we have selected the recently coined Fog-to-Cloud (F2C) computing paradigm [16], designed by other members of our research group, intended to create a hierarchically and distributed resources management system. Thus, we have designed all data management tasks during all data life cycle stages, from collection to removal, including storage and processing, on the F2C architecture. In addition, other important features, such as data quality and data transfer efficiency, have been also considered. The main advantages of the proposed model is that it combines the advantages of both, the cloud and the fog computing technologies, which mainly are keeping high performance capabilities for computational intensive applications, and reducing communication latencies for real-time or critical services, reducing network data traffic and enhancing fault tolerance and security protection.

There are several novel ideas in this proposal. First, we defined the COmprehensive Scenario-Agnostic DLC (COSA-DLC) model to manage and organize data in any scenario, science, or big data context. This model has been evaluated with respect to the 6Vs challenges to check its completeness and performance. We have also shown the easy adaption of this scenario agnostic model and have adapted it to the context of a Smart City, referred to as the Smart City Comprehensive DLC (SCC-DLC) model. Then, by designing and developing the SSC-DLC model on top of a F2C resources management framework, we benefit from the combined advantages of both the cloud and the fog computing technologies, these are: keeping high performance capabilities for computational intensive applications, reducing communication latencies for real-time or critical services, reducing network data traffic and enhancing fault tolerance and security protection. Finally, we have estimated the effects of our proposal and have evaluated the performance of some data management tasks in the Smart City of Barcelona, comparing our approach to the city centralized open data management platform.

1.4 Research Methodology

In this thesis, we have gone beyond the state of the art in the area of data management in specific context of smart cities. The research methodology used in the elaboration of this PhD thesis meets the traditional approach, we may structure into four global phases, as follows. In a first phase, we have reviewed the existing literature about DLC models, evaluated these models,

and concluded that there is no comprehensive DLC model suiting the specific needs for the mentioned smart cities context, thus setting the problem assessment. Aligned to this research need, in a second phase we have proposed a general data management framework, referred to as COmprehensive Scenario Agnostic DLC (COSA-DLC) model, intended to manage and organize data in any scenario, science, or big data context. This model has been evaluated in terms of the 6Vs data challenges, to validate its performance and completeness. Then, in a third phase, we show the COSA-DLC model applicability. To that end, we extend the existing COSA-DLC to work in the context of a Smart City, referred to as the Smart Cities Comprehensive DLC (SCC-DLC) model. Finally, in the fourth and last phase, we present a novel architecture for efficient fog-to-cloud data management in smart cities, consisting in mapping the SCC-DLC model into the Smart City F2C resources management architecture. The SCC-DLC consists of three main blocks, data acquisition, data processing, and data preservation. Data acquisition is mainly performed at fog layer 1, as well as some basic data processing and data preservation actions. The fog layer 2 can enhance the data processing and data preservations capabilities of level 1 by providing higher computing capabilities. And finally, the cloud layer is the responsible of a more complex and more sophisticated data processing over a much broader set of (presumably historical) data, as well as the responsible for permanent data preservation. These three blocks turns into the main chapters this manuscript is split into.

Indeed, the data acquisition block and its phases, collected the first design efforts for the SCC-DLC model, intended to show the ease adoption of several data aggregation optimizations in the proposed F2C data management model. For the sake of illustration, the Smart City of Barcelona has been deeply analyzed with the aim of estimating the amount of data generated by the sensors network, later collected and managed through its central platform, named Sentilo [17, 18], aiming to perform a data volume projection considering an exhaustive sensors development. We analyze the scalability of this eventual deployment and discuss different solutions to deal with this complexity. We have applied some data filtering and data aggregation techniques as part of data acquisition process.

Afterwards, the second block, data preservation attracts our research attention. We presented the advantages brought by storing data (including real-time, least-recent, and historical data) in a distributed hierarchal model, deployed by our F2C data management model. We have appraised the amount of available data to be stored in the storage media (from fog to cloud layers) at Barcelona Smart City. In addition, several challenges, imposed by storing the huge volume of data in different layers of fog to cloud –such as, retrieving real-time and historical data, the frequency of updating data, etc.–, came up. In this thesis, we propose an F2C object store service model, able to organize the writing model for the stored data in our F2C data management model.

The third block, the data processing is then analyzed, showing the benefits brought by our F2C data management model to ease the provision of services smart cities users. Our main interest focuses on showing how an F2C object store service model is able to read data for the services in our distributed F2C data management model. Finally, we highlight some details information about the network latency among layers in the F2C system.

1.5Structure of thesis

This thesis manuscript is in eight chapters as follows. Chapter 2 refers to the data concepts (including Big Data, Open Data, Open Data Government, and 6Vs challenges). In Chapter 3, we describe current Smart City trends. In Chapter 4 we adapt the COSA-DLC model to a Smart City scenario (SCC-DLC), proposing our initial F2C data management architecture. In Chapter 5, we dig into the Data Acquisition block with their related phases (including data collection, data filtering, data quality, and data description) and depict the F2C Data Acquisition block for the smart cities. In Chapter 6, we introduce the different phases in the Data Preservation block and describe the F2C Data Preservation block for the smart cities. In Chapter 7, we argue about the Data Processing block and their phases for smart cities (including data process and data analysis), and present the F2C Data Processing block for the smart cities. Finally, we conclude this thesis in Chapter 8, summarizing the research work performed and highlighting the main contributions of this thesis. In the following paragraphs the organization of this thesis is described in more detail.

Chapter 2:

Section 2.1: We define some important data concepts (including Big Data, Open Data, and Open Government)

Section 2.2: We argue about Vs challenges (including 3Vs, 4Vs, 5Vs, 6Vs, and so on) as part of Big Data complexities concepts.

Section 2.3: We describe concepts and issues about data management and review the definition of a Data LifeCycle model. Then, we show in detail, most existing Data LifeCycle models focusing on their challenges and limitations.

Section 2.4: We evaluate the presented Data LifeCycle models with respect to the 6Vs challenges.

Chapter 3:

Section 3.1: We introduce main sources and devices in the Smart City.

Section 3.2: We argue about main proposed technological architecture to connect sources and devices in the Smart City.

Section 3.3: We define the different computing models in the Smart City, including cloud, fog, and fog-to-cloud computing models.

Section 3.4: We go in details about resource management in the Smart City, encompassing centralized and distribute resource management.

*Section 3.5.*We describe data management in Smart City and then we highlight that a centralized data management (cloud data management) model is only existed in the current Smart City scenarios. So, we present that there is not any focus about distributed data

management (i.e. fog-to-cloud data management) model specially tailored to meet the smart cities needs, limitations and requirements.

Section 3.6: We depict a real Smart City scenario to use for our real experiments in the rest of this thesis. Our interest is the Barcelona Smart City as one of the big metropolitan city in the Europe.

Chapter 4:

Section 4.1: We propose a novel Comprehensive Scenario Agnostic Data LifeCycle (COSA-DLC) model, a DLC model which: i) is proved to be comprehensive as it addresses the 6Vs challenges (namely Value, Volume, Variety, Velocity, Variability and Veracity; and ii), it can be easily adapted to any particular scenario and, therefore, fit the requirements of a specific scientific field. We also include two use cases to illustrate the ease of the adaptation in different scenarios. We conclude that the comprehensive scenario agnostic DLC model provides several advantages, such as facilitating global data management, organization and integration, easing the adaptation to any kind of scenario, guaranteeing good data quality levels and, therefore, saving design time and efforts for the scientific and industrial communities.

Section 4.2: We present the Smart City Comprehensive Data LifeCycle (SCC-DLC) model, a data management architecture generated from a comprehensive scenario agnostic model, tailored for the particular scenario of Smart Cities.

Section 4.3 and Section 4.4: We define the management of each data life phase, and describe its implementation on a Smart City with Fog-to-Cloud (F2C) resources management, an architecture that combines the advantages of both cloud and fog strategies.

*Section 4.5.*We highlight summary and contribution of this Chapter.

Chapter 5:

Section 5.1: We present the data acquisition block includes the data collection, data filtering (which performs some optimizations, such as data aggregation), data quality (aiming to appraise the quality level of collected data), and data description (tagging data with some additional information) phases.

Section 5.2: We tailor the Data Acquisition block and their phases to the real F2C Smart City scenario.

Section 5.3: We estimate the network traffic in F2C data management model during data collection and compare it with a traditional real system.

Section 5.4: We discuss about summary and contribution of this Chapter.

Chapter 6:

Section 6.1: We introduce the data processing block encompasses the data process (which provides a set of processes to transform raw data into more sophisticated data/information), and

data analysis (implementing some analysis or analytic approaches for extracting knowledge) phases.

Section 6.2: We map the Data Preservation block and their phases to the real F2C Smart City scenario.

Section 6.3: We estimate the data storage size (including maximum, minimum and total capacity) in F2C data management.

Section 6.4: We highlight all summary and contribution of this Chapter.

Chapter 7:

Section 7.1: We show the data preservation block consists of four phases which are the data classification (aiming to organize and prepare data for efficient storage), data archive (storing data for short and long terms consumption), and data dissemination (publishing data for public or private access).

Section 7.2: We couple the Data Processing block and their phases to the real F2C Smart City scenario.

Section 7.3: We show all summary and contribution of this Chapter.

Chapter 8:

This chapter reviews and summarizes the proposed ideas of this Thesis. Moreover, it suggests avenues for future work.

Chapter 2:

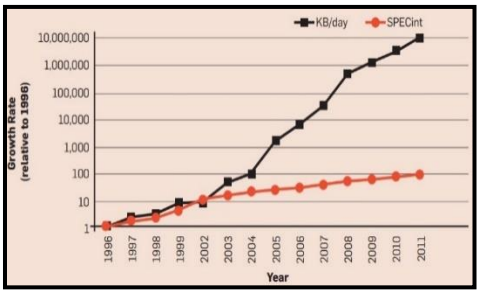
Data Management Concepts

This Chapter is structured as follows. Section 1 introduces the data generation concepts, focusing on insights related to Big Data, Open Data, and Open Government Data concepts. Section 2 describes the 6Vs challenges, to be used for evaluation purposes. Then, Section 3conceptualizes the Data Lifecycle model, reviews most relevant existing models, leading to their evaluation in terms of the 6Vs challenges in Chapter 4. In Section 5, we describe our proposal for managing data complexity, i.e., the comprehensive, scenario agnostic, DLC (COSA-DLC) model and evaluate it also in terms of the 6Vs challenges for benchmarking. Finally, in Section 6, we summarize the work done and open up avenues for further research.

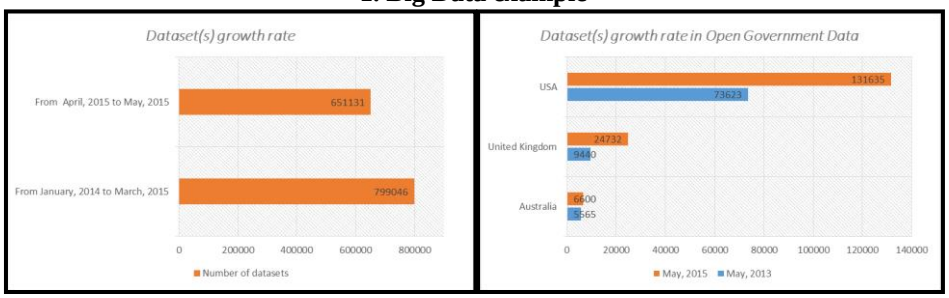
2.1 Data Generation concepts

Nowadays, the Information Technology (IT) world is quickly, continuously and unstoppably progressing towards unforeseen frontiers, empowered by Social Media, Internet of Things (IoT) as well as by emerging smart scenarios, such as smart cities, smart transportation or smart health. This IT progress is driving the need to handle a very huge volume of data as required by the set of services and devices building the envisioned IT smart scenario. As a consequence, a very large amount of data collected in either structured, semi structured or unstructured format (Big Data), are being stored in distributed data repositories to be shared and openly (Open Data) utilized by potential clients for either private or public usage (Open Government Data). However, it is widely accepted that sharing huge amounts of heterogeneous data brings some challenges yet unsolved, mainly related to an efficient and smart data processing. Some authors have already identified the set of challenges that must be addressed, coining the 5Vs challenges, namely Volume, Velocity, Variety, Value and Veracity.

We now introduce three examples to illustrate the growth rate curve for data production. The first example, refers to Big Data, and seats on the Next Generation Sequencing (NGS) [17, 18] in the biological systems field. As shown in Figure 2.1(Big Data example), the data growth evolved from a rate of 1 KB per day in 1996 to a rate of 10 GB per day in 2011. As a result, the size and number of experimental datasets available, from 1996 to 2011, keeps growing exponentially. The second example, refers to Open Data and is promoted by the Government of Catalonia [19, 20]. Figure 2.1 (Open Data example) shows the data collected by the app "Mobile Coverage", starting from January 2014 till now. The data are split into two periods, one lasting 15 months and the other one (the more recent) lasting two months. We may easily see how the magnitude of the last one is pretty close to the first, even though the 1 to 15 time relation, what undeniable shows the enormous growth in data. The third example, refers to Open Government Data, and focuses on data offered by three different countries (Australia, United Kingdom and USA) [21-23]. Figure 2.1 (Open Government Data example) also shows the datasets growth rate from May 2013 to May 2015. Again, we may easily observe a huge data evolution in both United Kingdom –more than twice from 2013 with 9,440 datasets to 2015 with 24,732 datasets–, and USA –an impressive 78,8% in the same period, from 73,623 datasets in May 2013 to 131,635 datasets in May 2015.



1. Big Data example



2. Open Data example **3. Open Government Data example**

Figure 2.1 Some examples of data growth rate from 1996 to 2015

Unfortunately but unavoidably, the continuous increase in the volume and diversity of data is adding high complexity and severe difficulties to all data life stages to be faced by technology stakeholders, particularly when dealing with real time data analytics. Thus, concepts such as Big Data, Open Data and the most recent Open Government Data are stressing the overall data life cycle processing, while simultaneously easing both users' access to data (leading to transparency, participation, and collaboration for customers [24]), and the deployment of new added-value services. It is worth emphasizing that by an easy users' access to data and the deployment of new services, the continuous development of the three mentioned concepts is strongly empowered, what undeniably sustains and even increases the overall data processing complexity.

For this reason, the worldwide scientific community has invested substantial efforts in the recent decades to overcome the challenges related to managing difficulty and complexity in the different aspects related to data life cycle (e.g. data collection, data processing, data analysis, data storing). To that end, the concept of Data Lifecycle model was formally defined, thus proposing different Data Lifecycle models as a high-level framework encompassing all data management aspects, from data creation to data consumption. The main goals for a Data Lifecycle model are: i) to eliminate waste; ii)to operate efficiently, and; iii) to prepare data products ready for end-users matching the expected quality constraints [25]. However, Data Lifecycle models are usually tailored to specific fields and interests, turning into particular goals and different data stages depending on the designer's needs for each data stage.

This Chapter goes deep into the main concepts related to the Data Lifecycle model, aiming at two concrete objectives. First, the Chapter surveys most of the existing Data Lifecycle models, particularly emphasizing the need for deploying a widely adopted solution not tailored to specific scenarios. Second, the Chapter points out the weaknesses of existing 5Vs challenges and

proposes a novel set of 6Vs challenges to evaluate most relevant Data Lifecycle models, highlighting limitations and weaknesses for each of them.

2.1.1 Big Data, Open Data, and Open Government Data concepts

This section deepens into Big Data, Open Data, and Open Government Data concepts, carefully analyzing pros, cons and main challenges for each initiative. In fact, having a solid knowledge about any ongoing initiative is crucial to get a comprehensive picture about the overall scenario but also to get the required background to facilitate the design and development of innovative solutions fixing the yet unsolved challenges.

2.1.1.1 Main Big Data concepts

Coined some years ago, the "Big Data" term [26, 27] has been largely defined by the data scientific community. However, although the technical relevance and business impact of Big Data is widely recognized, there is no global consensus on a uniform and highly accepted definition [28]. That said, after a thorough reading process on the current related literature we may conclude that Big Data definition scan be categorized into three blocks depending on the main characteristic used to formally establish the definition. The three semantic approaches can be classified as "data size", "technologies and processes", and "challenges". Table 2.1 highlights relevant references in the literature for each one of the mentioned characteristics:

Table 2.1. Big Data definitions

Characteristic	***Ref.***	***Description***
Data Size	[29]	Michel Cox and David Ellsworth were among the first to use the term big data literally, referring to the usage of larger volumes of scientific data for visualization.
	[30]	Big data refers to datasets whose size is beyond the ability of typical database software tools to capture, store, manage, and analyze.
	[31]	Big Data is about the growing challenge organizations face as they deal with large and fast-growing sources of data or information that also present a complex range of analysis and use problems.
Technologies and Processes	[28]	Big data shall mean the datasets that could not be perceived, acquired, managed, and processed by traditional IT and software/hardware tools within a tolerable time.
	[32]	Big Data and data intensive technologies are becoming a new technology trend in science, industry, and business.
	[27]	Big Data is a set of techniques and technologies that require new forms of integration to uncover large hidden values from large datasets that are diverse, complex, and of a massive scale.
	[26]	The process of handling big data encompasses collection, storage, transportation, and exploitation. It is with no doubt that the collection, storage, and transportation stages are necessary precursors for the

		ultimate goal of exploitation through data analytics, which is the core of big data processing.
	[33]	Big data is a term encompassing different types of complicated and large datasets, all becoming hard to process with the conventional data processing systems.
	[34]	Big data shall mean the data of which the data volume, acquisition speed, or data representation limits the capacity of using traditional relational methods to conduct effective data analysis or the data which may be effectively processed with important horizontal zoom technologies.
	[35]	Big Data is a massive volume of both structured and unstructured data, so large that makes difficult its to process using traditional database and software techniques.
Challenges	[27, 36-40]	The "V" discussion is started with 3Vs models. The terms 3Vs were originally introduced by Gartner to describe the elements of big data challenges, ending up characterizing Big Data by the three Vs, namely Volume, Velocity and/or Variety.
	[26]	Later, the 3Vs models were extended to 4Vs. However, there were some doubts on the added V. This reference includes veracity, so showing the 4Vs as volume, velocity, veracity, and variety.
	[28]	Another alternative for the added V is value, so defining the 4Vs as volume, velocity, variety, and value (instead of veracity).
	[32, 41, 42]	Recently, the discussion moved to 5Vs. Like 4Vs, there are different thoughts to characterize the 5Vs. Some references pushed for a 5Vs model including volume, velocity, variety, value, and veracity.
	[33]	This reference pointed out variability (replacing veracity), so defining 5Vs as volume, velocity, variety, value and variability.
	[43, 44]	New discussion goes to 7Vs as volume, variety, velocity, veracity, value, variability and visualization.

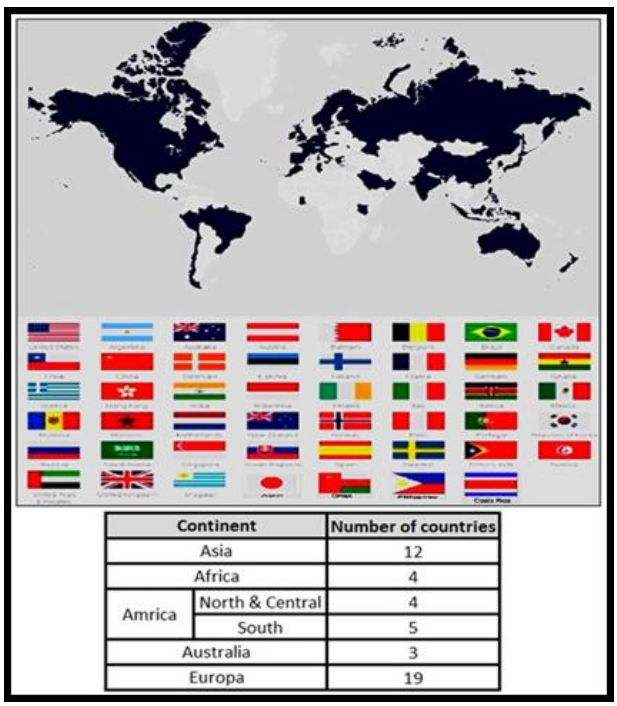
Big Data provides some opportunities and challenging effects. In short, most opportunities of Big Data refer to its economic impact [45], particularly dealing with optimizing production processes and supply chain, generating new goods and services, targeted marketing, improved organizational management as well as faster research and development. On the other hand, challenging effects of Big Data refer to the challenges highlighted by the exiting 5Vs challenges, mainly summarized into Volume (huge volume of data), Variety (various data formats), Velocity (rapid generation of data), Value (huge value but very low density) [28], and Veracity (quality and security of data) [32] which are later discussed in section 3.

2.1.1.2 Open Data

Many profit and nonprofit organizations establish public data spaces, generally referred to as Open Data [24, 45-49], enabling data sharing in a simple and elegant way in public or private spaces. In addition, data stakeholders send, publish and receive lots of different information through any connectivity technology in place. Thus, data come to open environments with distinct formats and sources by data stakeholders which make new challenges and opportunities for the information world. The main objective of Open Data is to provide data stakeholders with a public space for sharing information. Open Data is undoubtedly bringing in some benefits: i) business opportunities; ii) a free (or low-cost) public resource fostering innovation and a better-informed public[45]; iii) capacity to generate more services for users, and; iv) a more vibrant economy[47]. Nevertheless some weaknesses may also be reported, such as: i) the lack of data quality; ii) incompatible formats and access methods, or; iii) various semantic interpretations of data[48].

2.1.1.3 Open Government Data (Open-Data Government)

Several initiatives have been recently set by many governments around the world to create Open Government Data portals. In short, Open Government Data can be considered as Open Data provided by governments. This means that the data provided must have a valid and trusted reference for users as well as an acceptable quality. The aim of Open Government Data is to connect the customers to trustable and reliable information to benefit from better services. Its main rationale can be decoupled into a better governance, great and improved services, and a more vibrant economy [46, 50-55]. Forty-seven countries [56], as of May 2015, are already participating in the Open Government Data model throughout cities, states, and countries as shown in Figure 2.2.



Continent		Number of countries
Asia		12
Africa		4
Amrica	North & Central	4
	South	5
Australia		3
Europa		19

Figure 2.2 Countries joining the Open Government Data portals

Open Government Data portals bring huge pros and cons for Open Government Data stakeholders. On one side, major advantages of Open Government Data are transparency, participation, and collaboration [24], as well as valid sources and standardization of the data format. On the other side, the most notable barriers are the risk of violating existing regulations, difficulties with data ownership, misinterpretation and misuse of raw data, negative consequences of transparency and negative consequences for the government [57].

2.2 Extending the 5Vs challenges to a 6Vs model

The main challenges in Big Data have been traditionally described through the 3Vs challenges, Volume, Variety and Velocity, as defined by Gartner [27]. This model has been extended to the 5Vs challenges, which may be considered as 4 +1, since the latter differs depending on the reference. Indeed, 4Vs parameters include Value, Variety, Velocity and volume. The additional one may be either Variability as stated by [33, 58] or Veracity as read in [32, 41, 42]. Recently, there is some effort to show that the challenges can assume 7Vs [43, 44], including both Variability and Veracity, and adding Visualization as a new challenge. Some other authors propose Volatility, Viscosity and Virality as additional main challenges [59-73]; however, we think these are not mature enough to be considered in this work. Next, we introduce main references and concepts for all previous challenges among all sources.

- Value of data

  Value is a highest priority aspect of Big Data [27]. This is rooted on the fact that the main goal for data analysis and management, indeed, is to obtain enriched information. To reach this goal, we must explore large amounts of data with different data formats and sources to pick up some hidden data which can build the valuable information for business and end-users purposes [28]. The main challenge linked to this parameter refers to provide smartness approaches and scenarios for discovering and recognizing hidden value of information among all data.

- Volume of data

This parameter indicates the huge amount of data to be managed, coming from different data sources with distinct data types and formats. Note that Big Data concepts refer to the fact that a very large amounts of data must be managed and analyzed. Furthermore, traditional technologies can efficiently analyze and manage datasets limited to a certain size, typically in the size of Gigabyte. Therefore, the challenge is to promote new techniques and technologies to handle huge amounts of massive data, with different formats and sources [26, 28].

- Variety of data

Variety refers to the fact that data comes from different sources, like sensors, social networks, smartphones, and so on [27] and, therefore, the formats to be considered may be very diverse. These include structured, semi-structured, and unstructured data, such as audio, video, webpage,

plain text, etc. [28]. In addition, the continuous technology progress will definitely bring new devices and solutions enabling additional data collection (for example, we may envision highly impacting advances in the e-health or in the transportation sectors), what is also adding variability to the collected data. Thus, handling data heterogeneity is a relevant challenge for big data. Indeed, traditional tools, such as SQL, use to handle only structured databases, but do not perform well when the databases are semi-structured or unstructured [74].

- Velocity of data

Velocity highlights the extremely fast speed data streams are generated. For instance, sensors produce streams of data very fast, and the number of sensors are uncounted in smart scenarios. This demands a link between the Big Data concepts –specifically data collection, processing and analysis–, and a timely and efficient business value, for obtaining the expected value [26, 28, 74]. In addition, the business markets make plans for setting more frequent decision making closer to their customers' requirements. For instance, the bank industry needs to get online, or nearly online, data analytics –Fast Data analytics concepts–, to create better services for their customers in current days. Traditional algorithms and systems cannot manage the current data stream growth rate nor can process the increasingly growing data sizes [26, 39]. So, the main challenge is to propose appropriate technologies to satisfy the business requirements.

- Variability of data

Variability refers to the fact that data meanings can be changing and updating over the time. It refers to data semantic concepts which are related to the intrinsic and interpretations meanings of data [44]. For instance, sometimes one single word may stand for multiple meanings; or one word can be translated into different meaning depending on the sentence context; or even some words can change to different meanings throughout the time [43, 44]. This parameter is highly impacting stakeholders involved in data analysis [75]. The challenge points out that the definition of specific algorithms and approaches, like sentiment analysis and opinion mining, making text deeply and globally understandable are required [76].

- Veracity of data

Data veracity can be seen from two different points of view, namely quality concepts and security concepts, both defined next.

Veracity in Big Data, from a security perspective, guarantees that the data access will be secure, that is, unauthorized access and modification will be prevented. This makes data to be trusted, authentic and protected for end-usage [32]. The challenge is to guarantee that huge sets of data will be preserved against any unexpected change and attack during collection, processing, storing and any other stage during the whole data lifecycle [32].

Veracity in Big Data, from a quality perspective, guarantees that the data provided suits best end-users expectations [14]. The issue of data quality has been considered by a number of researchers, and includes topics such as data complexity, missing values, noise, imbalance, dataset shift and so on [77-81], and concentrates on details about how the data can be relied for making the best customers' decisions among all data [66, 82]. The challenge faces that despite the abundant data being available for usage, the quality of the data could be too complex for

decision making [26]. Furthermore, data quality can be guaranteed with two different strategies that try to prevent or correct errors throughout any activity [83, 84]:

- Quality Assurance (QA) is "*a part of quality management focused on providing confidence that quality requirements will be fulfilled*" regarding the ISO 9000 standard definition [85]. QA tries to prevent any kind of defect in the product with a focus on the process utilized to build the product [86].
- Quality Control (QC) is "*a part of quality management focused on fulfilling quality requirements*" regarding the ISO 9000 standard definition [85]. QC tries to identify and correct defects in the final products [83, 86].

- Visualization of data

Data visualization refers to the way data is presented, after being processed, as something easily visible, readable, understandable and tangible for most of the audiences like tables, diagrams, images and other intuitive display techniques [43]. The visualization may help users perform any potential analysis, seeking for possible solutions to endow business with better quality and performance. Such visualization however, is challenging since Big Data cannot be easily managed as traditional datasets. Indeed, some challenging issues in Big Data visualization are visual noise, information loss, large image perception, high rate of image change and high performance requirements [87].

After reviewing all definitions about the Big Data concepts, we may assess that, in our opinion, there is a difference between using the Vs model for Big Data definition and Big Data challenges. On one hand, the complete definition of Big Data can be generated considering variety, volume, and velocity, since these are the features that describe Big Data –value might be also included assuming Big Data has appeared fueled by the potential value among such massive data. On the other hand, with respect to the Big Data challenges, we do not believe that visualization, referring to a way of presenting data once processed [26], is one main challenge for Big Data technology. This is rooted to the fact that visualization is an optional software programming aspect for end-users. In this paper we propose the 6Vs challenges model, considering Value, Volume, Variety, Velocity, Variability and Veracity, as a model to evaluate the comprehensiveness of the different data lifecycle models considered in section 4.

Finally, in this thesis we propose the 6Vs challenges to be assumed for Open Data and Open Government Data concepts as well, since Open Data and Open Government Data both face the same challenges. However, the density of those challenges is different in Big Data, Open Data, and Open Data Government concepts. For instance, Open Government Data is less challenging in terms of Veracity assuming all datasets have been prepared with the unified sources; however some major challenges in terms of security and data quality under the Veracity challenge remain yet [57].

2.3 Data LifeCycle models

The chronological evolution about data generation, especially in terms of digital data, can be referred many years ago. But, it is with no doubt that the first digital data was created with the first advent of the computer generation worldwide (Data Creation). The generated digital data may be then eventually stored in different types of media for later usage (Data Storing). Afterwards, the stored data must be converted into data with sense and meaning, capable of being communicated or manipulated by some process (Data Processing). Thus, raw data can become information and useful knowledge through some specific processing, which basically garners data from a variety of sources that is then analyzed to feed end users with value-added benefits and advantages (Data Analysis). This basic roadmap, from data creation to data analysis through data storing and data processing, depicts a simple data life cycle. However, recognized the volume of data and the data formats heterogeneity, different Data LifeCycle models are currently positioned to manage and organize the data reacting to the specific needs, characteristics and requirements.

The Data management and organization concept is a critical and hot topic issue in any sciences (such as eScience), and big data environments (such as Smart Cities) intended to manage data sources, from creation to consumption in academia and industry nowadays. Traditionally, handling data management technologies is concentrated to the concept of Relational Database Management (RDBMS) and the more recent Extract-Transform-Load (ETL) process, for modeling data life stages in the context of data warehousing environments [8, 88, 89]. Recently, the emerging big data paradigm (in any scenario and science) has imposed further difficulties and complexities to the traditional data management systems [88]. Therefore, several DLC models are proposed as a set of planning, organization and management of data, from production to usage beyond any technology, system and software restrictions to overcome data management concerns [11, 12].

Data LifeCycle models are defined to set a high level framework standing for a global data life view from production stage to consumption stage. The benefits of designing and implementing a DLC model are the following:

- easing for planning and handling complexity of data management in all data life stages [11, 12];
- preparing data products ready for end-users, matching the expected constraints and efficiency [7, 11, 12];
- showing elucidate the quality level of data, removing any kind of waste and noise [11];
- illustrating a sequence of any essential activities related to data life [11]; and
- helping designers create sustainable software [90, 91].

Several proposals for Data LifeCycle models can be found in the literature as part of specific sciences and/or environments and/or data stages management, each of them addressing specific challenges and objectives about the particular science [12, 13], scenario [92, 93] or data stages management [94]. For instance, one simple and first Data LifeCycle model designed for collecting and storing data and managing specific curation policies, could be represented by three elements, namely Acquisition, Curation, and Preservation [11]. Acquisition refers to the

specific process of data collection. Curation stands for the process of preparing the collected data, as an initial feed, for a later utilization. Finally, Preservation refers to the process of keeping data available in any kind of physical source for future usage. However, it is worth mentioning that this naive model is to be meaningful for some particular scenarios requiring simple needs, such as collecting, filtering and storing data. Nevertheless, other more sophisticated models could be designed to meet additional users' data requirements and details, such as data plan definition, data quality control, security guarantees, or business policies to name a few. Thus, recognized the relationship between the end-user/data's requirements and the data challenges, the selection of the appropriate Data LifeCycle model for a particular problem is a key design step in the global data management strategy.

In the next Section we review most relevant Data LifeCycle models found in the literature, in order to provide readers with a solid and comprehensive picture of current efforts and trends done so far in this area. The objective is to illustrate gaps and weaknesses fueling the need for additional research. We will show that although several models have been proposed to manage data for specific scientific fields or individual projects, these models are customized to meet the set of end-users needs, hence only considering a few elements from a complete lifecycle of data. Other proposed models emphasize specific data phases, such as data curation or data preservation, hence only contributing on a narrow spectrum to meet individual challenges. In order to have an accurate but also broad evaluation, the completeness of the proposed Data LifeCycle models is evaluated on a broader basis defined by the set of 6Vs challenges introduced in [95].

2.3.1 Review of current Data LifeCycle models

This sections revisits a wide sample of existing Data LifeCycle models found in the literature. The objective for each model review is twofold, to describe the applicability context and to list the set of included elements, also highlighting the missing challenges.

2.3.1.1 The ANDS Data Sharing Verbs model

ANDS is an abbreviation of the Australian National Data Service. ANDS puts together information from data providers and publicly invested institutions, to be used by research institutions within Australia [92, 96]. The ANDS Data Sharing Verbs model aims to design systems for supporting data sharing and re-using. Create, Store, Describe, Identify, Register, Discover, Access, and Exploit are the steps defined in the Data LifeCycle model [94, 97-99]. This data model is focused to deal with enquires for information exchange within Australian research with simple access possibilities through an Internet based discovery [99].

Challenge: This model was created for a specific purpose, i.e., data sharing and data re-using. Therefore its applicability cannot be widely extended, hence it cannot be considered as a comprehensive or wide-applicability spectrum model. In addition, this model does not consider data quality, nor QA nor QC, what is also hindering its applicability and extendibility on those scenarios requiring such quality constraints.

2.3.1.2 The BLM model

The BLM (Bureau of Land Management) administrates the public lands in USA [100]. They propose a model for sharing information among customers with high quality levels. Plan, Acquire, Maintain, Access, Evaluate, and Archive are steps of the BLM model designed for land data management. This model has been designed as a non-linear representation, where QA and QC management are central issues. Hence, this model seems to work for data archiving and accessing with QA and QC management, as depicted in Figure 2.3 [101].

Challenge: The BLM model cannot be assumed as a comprehensive model because the orientation of the model is for information sharing with strong emphasis in data quality, only applied to a particular field, related to public lands information. Further, additional discussions are required on this model, mainly about time and cost efficiencies in this non-linear model, because quality assurance and quality control are nuclear components of this model.

Figure 2.3 The BLM model

2.3.1.3 The CSA model

The Cloud Security Alliance (CSA) is the world's leading organization working on security for cloud computing environments. CSA proposes a Data LifeCycle model for data security in the cloud environment, consisting in six phases, namely Create, Store, Use, Share, Archive, and Destroy, as shown in Figure 2.4 [102]. Again, this model only addresses one particular problem, i.e., security, for a specific scenario, i.e., cloud computing.

Challenge: The CSA model is not a comprehensive data model either, because it has been designed and customized for data security in cloud computing scenarios. This means that some concepts, such as data quality, data processing or data analysis are not considered, what unquestionably limits its extendibility.

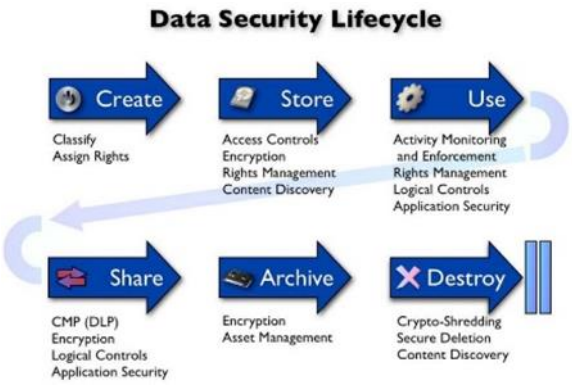


Figure 2.4 The CSA model

2.3.1.4 The DataONE model

The Data Observation Network for Earth is an organization, called DataONE, funded by the US National Science Foundation (NSF). Their data model aims to provide data preservation and re-using for research in biological and environmental sciences. The proposed Data LifeCycle model includes Collect, Assure, Describe, Deposit, Preserve, Discover, Integrate, and Analysis, as illustrated in Figure 2.5 [103, 104]. This particular model can be used for storing and retrieving information for long term usage.

Challenge: This model has been developed specifically for data preservation and re-using, which again limits its extendibility – for example, data security concerns are not included.

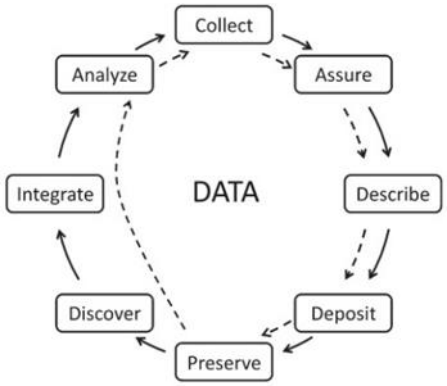


Figure 2.5The DataONE model

2.3.1.5 The DCC model

The Digital Curation Centre (DCC) is an organization addressing curation issues for digital information to improve higher education in the United Kingdom [105-107]. Indeed, DCC provides a model for successful data curation and preservation, assuming data in digital form. The proposed DCC LifeCycle introduces different layers: Full Lifecycle Actions, Sequential Actions and Occasional Actions. Full Lifecycle Actions are divided into four steps, namely Description and Representation of Information, Preservation Planning, Community Watch and Participation, as well as Curate and Preserve. Sequential Actions provide seven steps, namely Conceptualize, Create or Receive, Appraise and Select, Ingest, Preservation Action, Store,

Access, Use and Reuse, and Transform. Finally, Occasional Actions include three steps, namely Dispose, Reappraise and Migrate, as shown in Figure 2.6 [108, 109]. The steps disposition in this model is quite sophisticated as they are placed in a multiple layer cyclic structure. The main objective and focus for this model is on guaranteeing successful curation and preservation of digital data.

Challenge: This model is also designed for a particular scenario and objective hence impeding its wide adoption – for example, data analysis and data integration are not considered in this model. Moreover, the DCC model does not provide QA because the "Appraise and Select" step works like data QC in the LifeCycle [105].

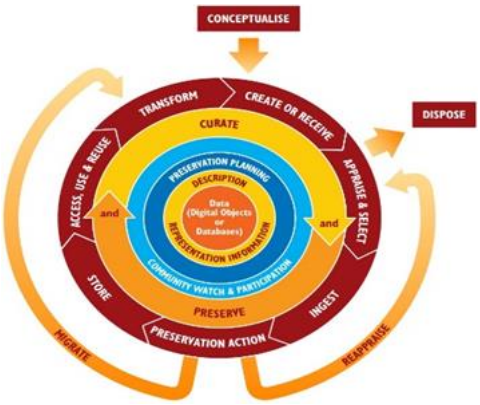


Figure 2.6TheDCC model

2.3.1.6 The DDI conceptual model, version 3.0

The Data Documentation Initiative (DDI) is a project of the Inter-University Consortium for Political and Social Research (ICPSR). The DDI aims at producing a metadata specification for the description of social science data resources. The offered model includes eight elements, shown in Figure 2.7, namely Study Concept, Data Collection, Data Processing, Data Archiving, Data Distribution, Data Discovery, Data Analysis, and Repurposing [92]. This model generates a conceptual model for political and social data research and standardization. In its version 3.0 the model provides the standardization for XML vocabularies.

Challenge: The DDI model successfully addresses most steps in the Data LifeCycle, from collection to consumption, what would ease its adoption as a comprehensive model. However, the model misses any approach for both data quality and security, what again limits its broad adoption.

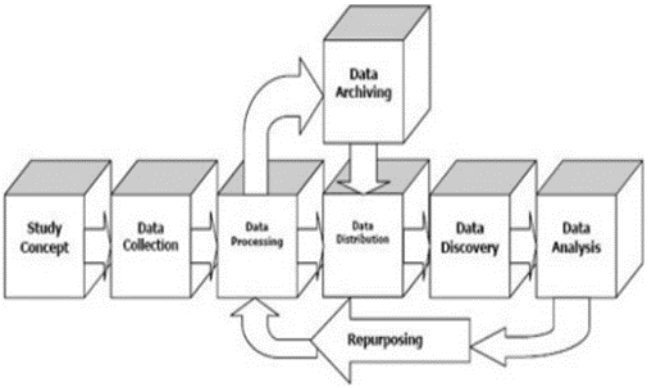


Figure 2.7The DDI Conceptual model, Version 3.0

2.3.1.7 The DigitalNZ Content model

DigitalNZ comes from Digital New Zealand and its goal is to collect and increase the amount of digital content for users. The data model is designed for both archiving and using the digital information. The proposed model includes Selecting, Creating, Describing, Managing, Preserving, Discovering, and Using and Reusing as steps, as shown in Figure 2.8 [110, 111], with the main goal of efficiently managing digital information exchange among data stakeholders.

Challenge: This model has been designed to focus only on archiving and using purposes, hence it cannot be considered as a comprehensive model either – for example, data analysis, data integration, data security and data quality steps are not addressed in this model.

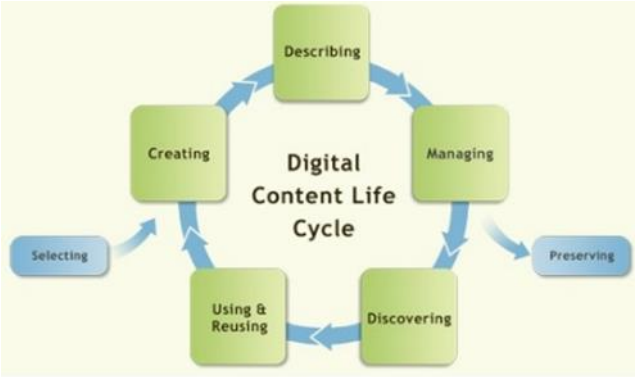


Figure 2.8The DigitalNZ Content model

2.3.1.8 The Ecoinformatics model

Ecoinformatics is a framework to help scientists work with the relevant biological, environmental and socioeconomic data and information. The data model aims to build new knowledge through creative tools and approaches for discovering, managing, integrating, analyzing, visualizing and preserving relevant data and information. As depicted in Figure 2.9, Plan, Collect, Assure, Describe, Preserve, Discover, Integrate, and Analyze are the steps included in the Ecoinformatics model [12, 13]. Hence, the model provides a framework to achieve new insights about data and information for some particular sciences.

Challenge: This framework design is close to be widely adopted since it copes with most relevant challenges, i.e., data collection, data preservation, data discovery, and some data manipulation, such as data integration and data analysis. However, data security is still an open challenge not included in the model, what unquestionably limits its extendibility. This model is similar to the DataONE model described in Section 3.4, only differing on the first step.

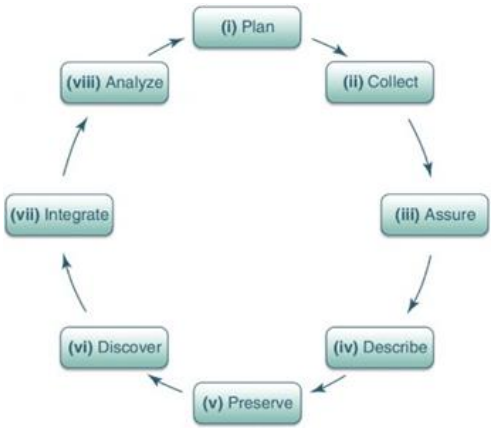


Figure 2.9The Ecoinformatics model

2.3.1.9 The Generic Science model

The Generic Science model is offered by the Science Agency in order to manage scientific digital data. There are six steps included in the Generic Science model, namely Plan, Collect, Integrate and Transform, Publish, Discover and Inform, and Archive or Discard. The model, shown in Figure 2.10, can predict the next set of data acquisitions with specific techniques to use for data management plans [101].

Challenge: This model is specifically designed for data archiving and disposing, hence it cannot be considered as a global comprehensive model – for example the model does not consider data analysis, data security or data quality.

Figure 2.10TheGeneric Science Model

2.3.1.10 The Geospatial model

The Geospatial Data LifeCycle model is supported by the Federal Geographic Data Committee (FGDC). The model aims to explore and save valuable information about geographic and spatial-related data activities. Figure 2.11 summarizes the Geospatial Data LifeCycle steps, i.e., Define, Inventory/Evaluate, Obtain, Access, Maintain, Use/Evaluate, and Archive [101, 112]. This model is proposed to discover data with acceptable quality and business requirements for future use.

Challenge: As the previous one, this model has been designed for a particular objective, specifically for searching and archiving information. Therefore the model cannot be positioned to be widely used in different scenarios – for example, the model does not address data analysis and data integration. Moreover, in this model QA and QC are included in each step, which can be a limitation in terms of runtime and work efficiency.

Figure 2.11 TheGeoSpatial model

2.3.1.11 The LOD2 Stack model

LOD2, the Linked Open Data, is a large-scale integrating project co-funded by the European Commission within the FP7 Information and Communication Technologies Work Program [42]. The LOD2 Stack data model looks for useful data matching the end-user requirements. This model includes Storage/Querying, Manual revision/Authoring, Interlinking/Fusing, Classification/Enrichment, Quality Analysis, Evaluation/Repair, Search/Browsing/Exploration, and Extraction as the different steps, as shown in Figure 2.12 [92, 113, 114]. In short, this model is helpful to find relevant data for end-users.

Challenge: The main objective for this model is to look for particular data, thus not useful for different objectives – for example, data security is not considered. Moreover, it is also worth noticing that the LOD2 Stack model manages quality, but just partially since only QC is managed – only the quality of the web contents is measured [113].

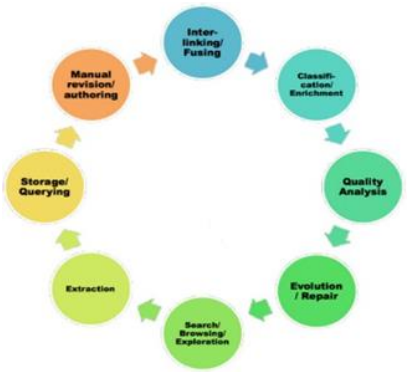


Figure 2.12 LOD2 Stack Model

2.3.1.12 The University of Deusto model

A group of researchers from the University of Deusto, in Spain, have proposed one Data LifeCycle model for data management in smart cities [92]. As depicted in Figure 2.13, the different steps of this model are Discovery, Capture, Curate, Store, Publish, Linkage, Exploit and Visualize. This model looks to be a proper candidate for discovering, storing, and publishing data in smart cities.

Challenge: This model focuses on smart cities scenarios, thus too narrow for wide adoption. In addition, the model does not focus on data security, nor on data quality (including QA and QC).

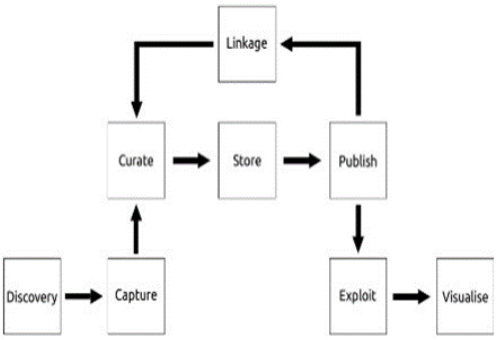


Figure 2.13The University of Deusto model

2.3.1.13 The Records model

The University Archives and Historical Collections (UAHC) department assists the Michigan State University to grant an efficient data administration and management, meeting the severe university's procedures. The Records data model aims to offer a solution for moving paper work to digital work in any kind of scenarios, especially in the university. The offered model includes different steps, i.e. Create/Receive, Use and File, Transform and Store, Dispose and Archive/Destroy, as shown in Figure 2.14 [101, 115]. In summary, the model provides an electronic procedure for making more efficient and better administration and management in the university.

Challenge: The Records model is not extendable, since it was designed to focus on data archiving – for example, the model does not consider data quality, data analysis, data processing, and data integration.

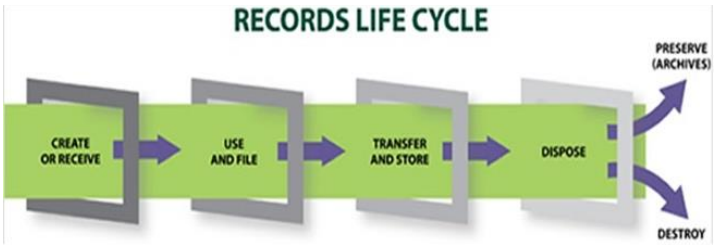


Figure 2.14 The Records model

2.3.1.14 The JISC Research model

The Joint Information Systems Committee (JISC) under the Managing Research Data Program, works for efficient research data management and sharing for the UK Higher Education and Research. The proposed Research model has been designed for discovering and sharing data among users. The model includes seven steps, namely Plan, Create, Use, Appraise, Publish, Discover, and Reuse, as drawn in Figure 2.15 [110, 116].

Challenge: This model cannot be widely adopted because it has been particularly designed for data sharing and discovery – for example the model does not offer any step for data processing, data integration or data analysis. Moreover, the model covers QC concepts within the "Appraise" step, but QA is not provided.

Figure 2.15 The JISC Research model

2.3.1.15 The UK Data Archive model

The UK Data Archive works among the largest collection of digital data, including social and economic data, in the United Kingdom. The UK Data Archive model focuses on acquisition, curation and archive of the digital data. To that end, the proposed model includes Creating Data, Processing Data, Analyzing Data, Preserving Data, Giving Access to Data, and Re-using Data, as shown in Figure 2.16 [92, 97]. In summary, the model can be a good choice for archiving and discovering across the digital data.

Challenge: This model could be considered as a comprehensive model because it provides the full Data LifeCycle, which includes acquisition, curation and preservation. However, the model concentrates on particular social and economic sciences, what limits its broad adoption. Moreover, the model does not cover data quality issues.

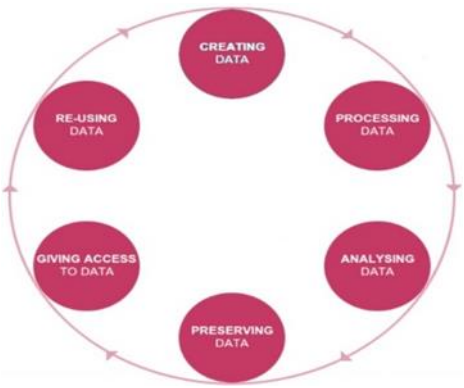


Figure 2.16The UK Data Archive model

2.3.1.16 The USGS model

The U.S. Geological Survey (USGS) Community for Data Integration (CDI) works with data and information management issues that can be relevant for the U.S. Bureau's Scientific Research. The USGS data model provides a framework to evaluate and improve policies and practices for managing scientific data, as well as to identify areas in which new tools and standards are needed. The model includes Primary and Cross-Cutting model elements, as depicted in Figure 2.17 [15, 25]. The Primary model elements are Plan, Acquire, Process, Analyze, Preserve, and Publish/Share. Besides that, Cross-Cutting model elements come with Describe, Manage Quality, and Backup and Secure, as steps. Thus, this model can be a reference to manage the scientific data for better standards and tools.

Challenge: Similar to the previous one, this model could be considered as a comprehensive model because it suggests data cycles for acquisition, curation and preservation. However, this model chooses a linear presentation for the graphic model, so time and work efficiency should be under discussion, especially for large amounts of data. In addition, the model does not cover data security because the meaning of secure in the "store and secure" element refers to physical risk, such as hardware and software failures [15].

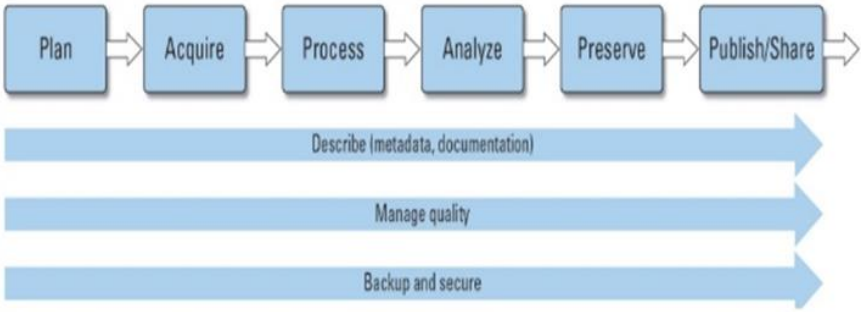


Figure 2.17The USGS model

2.3.1.17 The Beijing University model

The model comes up from a research team at the Beijing University on Posts and Telecommunication, in China. This model is used for data security in the cloud computing environment. The graph-based model, depicted in Figure 2.18, includes five steps, namely Create, Store, Use and Share, Archive, and Destruct [117]. This model is appropriate for security data in cloud environments.

Challenge: This model cannot be considered as a comprehensive model because it is designed only to support data security in the cloud – the model does not consider data quality, data analysis and data publishing in any of the proposed steps.

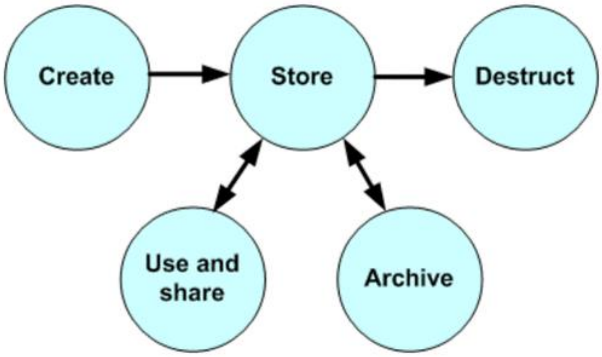


Figure 2.18The Beijing University model

2.4 Evaluation of the Data LifeCycle models

Section 3 analyzed and deeply described most Data LifeCycle models found in the literature. Our focus was oriented to two key aspects. First, although several Data LifeCycle models have already been designed, they are tailored to solve particular problems or areas in data management in (several) science(s). It is with no doubt that the proposed solutions may be complete in the set of challenges they cover, but unfortunately the proposed solutions follow a narrow-design strategy, only addressing the specific challenges for the particular problem, hence impeding its extendibility to other wider scenarios. Thus, these solutions are not wrongly designed but too customized to individual scenarios.

Second, the fast and unstoppable technological evolution is fueling the development of new services and apps, bringing new opportunities not only for traditional providers (telcos, services developers, etc.) but also for users that are easily adopting new technologies to accommodate their needs, thus playing new roles in the overall market (for example, prosumers, i.e. consumer + producer). This evolution turns into a smarter IoT world, where traditional areas, such as transportation or health, are enriched with smart capabilities leveraging technology evolution but also data availability. The coming IoT world is envisioned as a set of high-impacting services (user-transparent and technology-agnostic) deployed for and by users to improve users' quality of life (in different areas). A successful deployment of this IoT world is strongly dependent on the available data and on the processing capacities required for data management, hence an accurate, fast, reliable, secure, etc., global and widely deployed model for data (mobile, diffuse, un-structured, heterogeneous, etc.) management is demanded and undoubtedly mandatory.

For this reason, we analyze whether there is a global and comprehensive Data LifeCycle model already in place, which can successfully deal with most challenges required to support the coming IoT world demands. In short, the main contribution of a comprehensive model is to eliminate waste and duplicity in researchers' tasks to design a new model for any new project. To that end, in this section, we evaluate all introduced Data LifeCycle models with respect to the 6Vs challenges proposed in [95], including Value, Volume, Variety, Velocity, Variability, and Veracity, in order to know to what extent each model is comprehensive enough to be widely adopted as a general purpose model. Table 2.2 shows the results of the Data LifeCycle models

evaluation with respect to the 6Vs challenges. The evaluation is marked "yes" (□) at each box if the model can handle the corresponding V challenge, otherwise it is marked "no" (x).

In order to complete the table, we have assumed that, as volume and variety are fundamental challenges in Big Data management, all Data LifeCycle models are able to address both of them. For that reason, we have marked "yes" in the corresponding table column for all models. In addition, all Data LifeCycle models have been designed to manage data in specific environments, what means that only some challenges will be addressed on each model. For example, the CSA model has been proposed to provide security for Cloud computing environments. Therefore, this model is appropriate to be considered in environments where volume and variety of data has to be managed, so we mark "yes" for volume and variety. But value, velocity, variability, and veracity must be deeper reviewed. Eventually, we have marked "yes" in security, as part of veracity, because this model is specific for data security, but we have marked "no" for variability, velocity, and QA and QC, as another part of veracity, because in its description there is no clue on how these challenges would be handled.

From Table 1 we can conclude that: i) data quality –as part of veracity–, and velocity, are challenges that have only been considered in very few models, hence a comprehensive Data LifeCycle model must pay more attention to these challenges, so guaranteeing data quality and fast data generation, as important keys in Big Data management, and; ii) there is no Data LifeCycle model completely covering all the 6Vs challenges within the LifeCycle steps. The USGS model gets closer to this completeness; however, it still lacks of data security. Therefore, we may conclude that there is no global and comprehensive model with respect to the 6Vs challenges in this evaluation.

Table 2.2 Make evaluation of the Data LifeCycle models

	Data LifeCycle models	6Vs Challenges							
		Value	Volume	Variety	Velocity	Variability	Veracity		
							QA	QC	Security
1	ANDS Data Sharing Verbs model	x	☑	☑	x	x	x	x	☑
2	BLM model	☑	☑	☑	x	x	☑	☑	☑
3	CSA model	x	☑	☑	x	x	x	x	☑
4	DataONE model	x	☑	☑	x	☑	☑	☑	x
5	DCC model	☑	☑	☑	x	x	x	☑	☑
6	DDI conceptual model, version 3.0	☑	☑	☑	☑	☑	x	x	x
7	DigitalNZ Content model	x	☑	☑	x	x	x	x	x
8	Ecoinformatics model	☑	☑	☑	x	☑	☑	☑	x
9	Generic Science model	☑	☑	☑	x	x	x	x	x
10	Geospatial model	☑	☑	☑	x	x	☑	☑	☑
11	LOD2 Stack model	x	☑	☑	☑	☑	x	☑	x
12	University of Deusto model	x	☑	☑	☑	☑	x	x	x
13	Records model	x	☑	☑	x	x	x	x	x
14	JISC Research model	☑	☑	☑	x	x	x	☑	☑
15	UK Data Archive model	x	☑	☑	☑	☑	x	x	☑
16	USGS model	☑	☑	☑	☑	☑	☑	☑	x
17	Beijing University model	x	☑	☑	x	x	x	x	☑

Chapter 3:

Smart City Concepts

The world human population is estimated to increase from 7,336 million in mid-2015 to 9,804 million in mid-2050 [27]. This human population evolution is and will be demanding more and much better services aiming at city resources optimization. Smart Cities provide new solutions and opportunities for efficient management through the use of the most advanced Information Technology. In fact, it is widely accepted that Smart Cities have the potential to improve its citizens' quality of life through the economy development, the social and political progress, the availability of new services and solutions, the protection of the environment, the hyper connectivity among citizens, and so on [118]. In addition, the Smart City concept can be applied in several domains, such as smart environment, smart energy, smart transport, smart health, or smart security, just to name a few.

Nowadays, Smart Cities are positioned as one of the most challenging and important research topics, highlighting major changes in people's lifestyle. Currently, several definitions have been proposed to define the "Smart City" term, some of them listed next [119]:

- IBM in 2010 defined "*the use of information and communication technology to sense, analyze and integrate the key information of core systems in running cities*".
- In 2011, Caragliu mentioned that "*a city is smart when investments in human and social capital and traditional (transport) and modern (ICT) communication infrastructure fuel sustainable economic growth and a high quality of life, with a wise management of natural resources, through participatory governance*"

Technologies such as smart energy, smart transportation or smart health are being designed to improve citizen's quality of life. Smart Cities leverage the deployment of a network of devices – sensors and mobile devices–, all connected through different means and/or technologies, according to their network availability and capacities, setting a novel framework feeding end-users with an innovative set of smart services. Aligned to this objective, a typical Smart City architecture is organized into layers, including a sensing layer (generates data), a network layer (moves the data), a middleware layer (manages all collected data and makes it ready for usage) and an application layer (provides the smart services benefiting from this data).

Indeed, smart cities goals are to upgrade the traditional services towards modern services supporting users' demands through high level technologies. This view objective is not only to citizens' prosperity, but also to economic progress and sustainability of the city. In addition, cities compete and invest many efforts and budget to endow their citizenships with innovative services. Several reports classify cities around the world ranking smarter cities according to different factors, such as energy efficiency, transport effectiveness, public management, and so on. Recently, IESE Cities in Motion Index (CIMI) 2015 reported that Barcelona is ranked first in Spain and thirty-fourth in the world [120, 121]. The report has analyzed some values, such as human capital, social cohesion, public management, the environment, mobility and transport, technology, or the international outreach of the cities.

In this Chapter, we open some description about organizing and managing technologies in the city. First, we highlight the fact that a significant number of sources and devices (such as sensors) is deployed in a Smart City, to be used to sense the daily life in the city. In addition, each source

will produce data as the initial feed for Smart City services in the city. Second, we show how sources and devices can be communicated each other through a technological architecture in the city. Third, we describe how raw data can be converted into meaningful information by using computing models. This, different computing models exists intended to convert the raw data into information in a Smart City. Next, we explain some strategies about resource management in the Smart City. Those strategies go beyond managing resources in a city through centralized and decentralized models. And, we highlight that the advent of big data imposed new challenges for technological architecture, resource management and computing models in smart cities. So, we may conclude that the data management strategy in the city can coordinate this challenge. We also talk about some open challenges and directions for future smart cities. Finally, we dig into the real Smart City scenario in Barcelona city, focusing on a Smart City architecture for Barcelona, specifically on the sensor deployment and the Sentilo platform.

This Chapter, is structured as follows. Section 1 presents the most remarkable sources and devices in the smart cities. Section 2 shows the communication technologies in the smart cities. Section 3 describes the computing models in the smart cities. Section 4 talks about the resource management in the smart cities. Section 5 digs into data management models in the smart cities. Finally, Section 6 illustrates the Barcelona Smart City.

3.1 Sources and devices in Smart City

Nowadays, a large number of sources may be found in any Smart City. While in Chapter 5, we will deeply discuss different type of sources and devices, in this Section we present some of the main sources for data generation (including sensors, smart devices, and web services) in smart cities as shown below:

- The sensor is a physical device able to sense and get a specific type of input, such as light, heat, motion, pressure, or noise, from the physical layer. The input can be either converted into human-readable display at the sensor location, or transmitted electronically through the network for future use. There are many different types of sensors, for instance, acoustic and sound sensors (e.g. microphone), automotive sensors (e.g. speedometer), chemical sensors (e.g. pH sensor), electric and magnetic sensors (e.g. metal detector), environmental sensors (e.g. rain gauge), optical sensors (e.g. wave front sensor), mechanical sensors (e.g. strain gauge), thermal and temperature sensors (e.g. calorimeter), proximity or presences sensor (e.g. Doppler radar), and so on [122]. There are some important challenges for selecting and installing the appropriate sensors in a given context, such as accuracy, environmental condition, range, calibration, resolution, cost and repeatability [119].
- Smartphones are common devices spread all over the city that may behave as a great physical and mobile source for data gathering, everywhere and every time. It provides great opportunities to invest efforts for collecting data from anywhere at a very low cost. In fact, many smartphone apps are already sensing and saving large amount of data together with the user's location. In fact, some mobile applications are collecting such data

even without Internet connection and, therefore, this information is stored offline and can be retrieved later.

- Web services generate huge amounts of data that may be of interest in a Smart City environment. However, the veracity and quality of this information should be carefully considered before being used. Some efforts have been made to filter this data for a proper utilization in the context of a Smart City. Techniques such as Semantic Web, Linked Data, and so on [92], are used to obtain information from websites to the Smart City's stakeholders.

3.2 Connecting sources and devices in the Smart City

In a modern city there is an ever unlimited amount of resources and technologies, including computing devices (from smartphones, computers in vehicles, embedded computers, to personal computers or more powerful data centers), other devices to generate data (sensors in the city, sensors in users' devices, surveillance cameras, and so on), communication networks (wired networks, such as Ethernet, optical fiber, or wireless technology, such as 4G, WiFi, RFID, Bluetooth, or any other ad hoc networking technology), and several management platforms to facilitate and optimize users' interaction with the Smart City.

The Smart City technological architecture is typically organized into layers, including the sensing layer, the network layer, the middleware layer and the application layer [123]. The sensing layer consists of a broad network of sensors spread across the whole city, responsible for collecting as much data as possible through either different types of sensors, other smart devices, such as smart phones or smart vehicles, or also through web services, surveillance cameras, or GPS devices. The network layer connects the sensing layer to the middleware layer through a diverse set of communication technologies, such as cellular networks, satellite networks, WiFi, Ethernet, or any other ad-hoc technology enabling non easy to reach location connection. The middleware layer contains the main framework aimed at both organizing and centralizing the data collected as well as providing a platform for an easy and usually open access to the information. And eventually, the application layer provides the set of appropriate services for citizens or third party consumers [124].

Providing appropriate architectures for Smart Cities assuming different scenarios and environments has recently become an active research topic. There is no unique solution, rather several exist following similar patterns. In [118], Kyriazopoulou identifies six different architectural approaches to design a Smart City, such as Architectural Layers, Service Oriented Architecture, Event Driven Architecture, Internet of Things, Combined Architectures, or Internet of Everything.

Smart cities design through Architectural Layers (AL) defines a framework organized by some specific hierarchical layers. Each layer has a different responsibility and defines an interface to interact with higher and lower layers. The main aim of an AL framework is to offer an efficient solution for developing modular services and applications at each layer. Main AL architecture

advantages are both its simplicity and its modular design facilitating extensions for future development.

The Service Oriented Architecture (SOA) approach defines a platform for interaction between service stakeholders and services. Three roles are usually defined in SOA: the service provider, the service agent and the service consumer, to make the interaction between end-users, services and service provider through data collection, data filtering, and so on [125]. The benefit of the SOA architecture is its capability for adaption between service stakeholders and services.

The Event Driven Architecture (EDA) approach defines an architecture for managing asynchronous events under uncertain conditions. This architecture proposes a model to manage events in terms of creation, identification, utilization and response, which can, for instance detect emergency cases through sensors data registering. An advantage of an EDA model is its capacity to provide sustainability and security in Smart City environments through events management.

The Internet of Things (IoT) approach defines a framework enabling heterogeneous devices management, including sensors, smart devices, web services, and so on. Each kind of heterogeneous devices can connect and communicate with other through Internet, or any other local networks. In addition, Cloud Computing can also be used to help share computational resources also offering services to devices through Internet. Therefore, the IoT architecture creates an appropriate unified scenario to overcome all possible requirements for IoT stakeholders. The main profit of an IoT architecture is to provide connectivity with many different type of sources and provide variety of services for end-users.

The Combined Architecture is a new trend in current Smart City design that properly combines different aspects of the previous proposals, hence taking the most out of any individual architecture. Some popular architectures are IoT-SOA, IoT-SOA-AL, IoT-EDA, or IoT-SOA-AL-EDA.

The Internet of Everything (IoE) is a new generation paradigm to extend the IoT. Although no much difference is reported between both concepts, IoE could be understood as an IoT extension guaranteeing “everything” connectivity. The IoE proposes a new and broader paradigm for smart objects that can connect to each other easily and quickly, anytime and anywhere around the city. The highly increasing interest of the IoE architecture is to manage a wider variety of information, from any device, to create smarter applications and services through faster Internet infrastructure and network connection.

3.3 Computing models in Smart City

As we have discussed so far, many data exist in a Smart City coming from different sources and devices. These (raw) data are not meaningful for user demands. So, the first step is to transfer this raw data into meaningful information by an appropriate computing. In this Section, we present some of the computing models candidate to be deployed in smart cities (including cloud, Fog, and F2C computing) as shown in below:

3.3.1 Cloud Computing

Cloud computing has driven a global shift for computer processing, storage, and software delivery away from the desktop and local servers, across the network, relying on next generation data centers hosted by large infrastructure companies (such as Amazon, Google, Yahoo, Microsoft, or Sun) [126]. Definitely, cloud computing technologies provide extremely powerful computing resources over the network. Moreover, cloud computing offers the pay as-you-go model, where infrastructure is maintained by its owner, turning into a model very attractive for many companies (especially for start-ups and medium-sized businesses). In this way, cloud computing platforms (like those offered by Amazon Web Services, AppNexus, GoGrid, etc.) work differently than application service provider (ASP) and database-as-a-service (DaaS) paradigms. Instead of owning, installing, and maintaining the database software for a specific client (often in a multi-tenancy architecture), cloud computing stakeholders support little more than the hardware, and provide a set of virtual machines so that their customers may install their own software. In short, cloud computing offers resources (almost a seemingly infinite amount of computing power and storage) to be accessed by potential users in a pay-only-for-what-you-use pricing model. Indeed, cloud computing provides computing resources (including hardware, application development platform, and computer applications) available as services across the Internet, through three different schemes, Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [127].

Cloud computing has some desirable advantages, namely cost efficiency, reducing of system administrative functions, enhancing flexibility, increasing reliability and location independence. However, some undesirable disadvantages of cloud computing are high and unpredictable network latencies, difficulties in securities issues, critical and real-time applications requirements [127].

3.3.2 Fog Computing

Alternatively, Fog computing has been proposed to deal with some cloud weaknesses, for example, latency or high traffic load. The Fog computing boils down to extending the cloud towards the edge, thus putting together billions of Internet-connected devices into a distributed infrastructure, located at the edge of the network[128]. Accordingly, fog computing brings additional infrastructure (i.e., compute, storage, and networking) for services execution, located at the so-called fog-layer, somewhere between the edge and the cloud (traditional datacenters). Indeed, from a service execution perspective, Fog computing can provide substantial advantages for services execution in several domains (including Smart Grid, wireless sensor networks, IoT and software defined networks, SDNs).

The main objective of Fog computing is to offer computing (storage and network) resources closer to end-user devices, thus reducing both response time and network load, while also diminishing the security gaps. Unfortunately and obviously, the Fog infrastructure does not have powerful computing capacities as cloud, what along with the specific constraints and limitations inherent to the components deploying the fog layer, will pose several limitations to fog

deployment and utilization. As a consequence, it is reasonable to say that the main goal of Fog Computing is not to compete with cloud computing [129], but to complement each other.

As a summary, we may undoubtedly assess that Fog computing provides several benefits, such as lower latency, lower network traffic, location awareness, widespread geographical distribution, mobility support, the predominant role of wireless access, improving quality of services for streaming and real-time applications, scalability, heterogeneity, the orchestration of largescale control systems, hierarchical networking, and computing structures [23, 130, 131].

3.3.3 Fog-to-Cloud Computing

The Fog-to-cloud (F2C) computing model comes up to make the most out of combining Cloud and Fog computing models. In short, F2C is intended to use both advantages of Fog (close to users but limited resources) and cloud computing (far to users but unlimited and powerful resources). In addition, F2C Computing aims to enhance integration of Fog Computing and Cloud Computing through a coordinated management of underlying resources, while also bringing in innovative execution models, such as parallel and distributed execution of services into distinct fog or cloud resources[132]. The main rationale is to execute services on those resources best suiting the services needs, be it at cloud, at fog, or a combination of both [133].

In [134], the author proposed a simple three level hierarchy as depicted in Figure 3.1. We may see that, Fog devices can connect to the edge devices and also to Cloud.

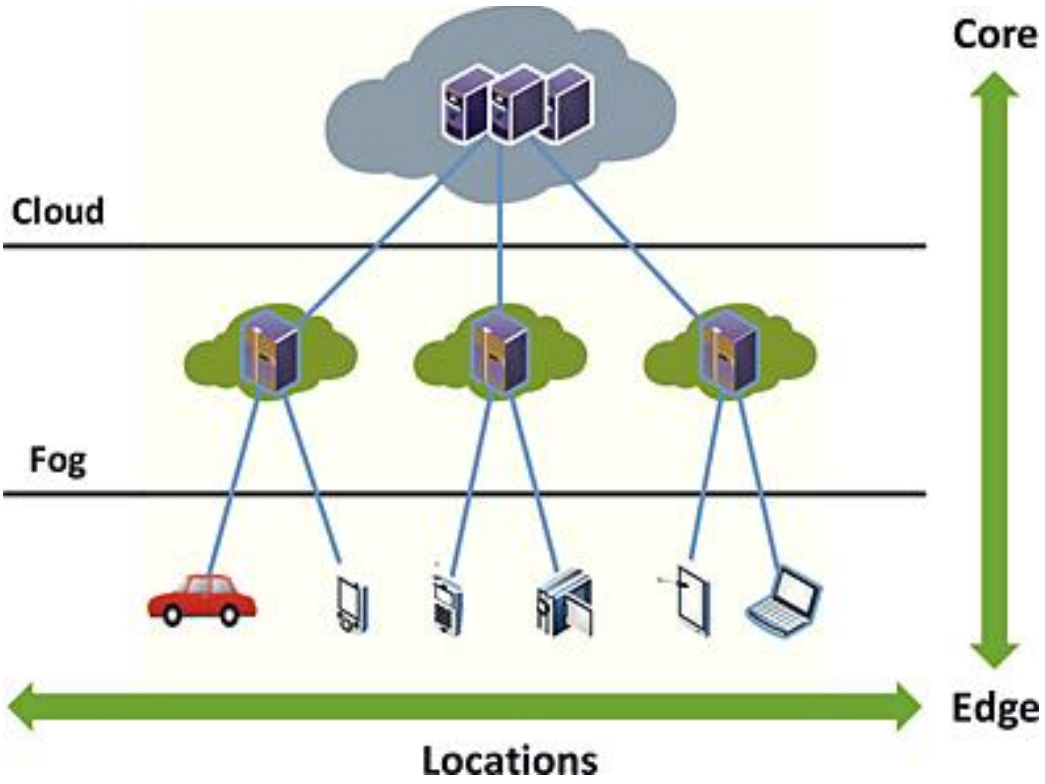


Figure 3.1 The basic F2C scenario [134]

3.4 Resource management in Smart City

Resources management in the context of smart cities can be approached from two main perspectives: centralized or distributed. In a centralized approach, a main data center (probably in the cloud) is the responsible for organizing and managing all resources from the city, gathering all data generated by sensors at the edge (traffic monitoring, energy meters, noise detection, or air pollution control, among others) and transferring them through some sort of global wireless communication technologies, such as 3G or 4G. In addition, processing facilities are also provided in the centralized data center, as long as it has very high computing and storage capabilities as shown more details in following.

3.4.1 Centralized resource management

Traditional resource management architectures in smart cities rely typically on centralized cloud computing facilities. Advantages of cloud computing are the (almost) unlimited computing capacity, the cost efficiency (market scale) and the elasticity (pay-as-you-go model) [135, 136]. However, moving all data and services to the cloud, which presumably may be far from the user, undoubtedly adds several inconveniences, such as high communication latencies, network overloading, and also increases the risks for failures and for security vulnerabilities [137-139].

For instance, Figure 3.1 shows an architecture for Smart City resources management based on cloud computing [140]. This model considers four layers, namely physical, network, cloud, and application layers. The physical layer includes all physical devices to obtain raw data from the city. The network layer provides support for sending the sensed data to the main cloud computing environment. The third layer is the cloud layer, which is able to process, compute and analyze all raw data, turning it into meaningful information mandatory for services execution. And the last layer is the service layer, which is ready to accessing data from the cloud layer and convert, interpret or combine each other for services and applications. In such scenarios, there is no doubt about the almost unlimited computing capacities and the ubiquity of such resources; however, some limitations due to the physical distance between resources and services can be reported, such as network overloading, high communication latencies, as well as high probability of failures and security risks, as mentioned before [138, 139].

3.4.2 Distributed resource management

Alternatively, in a distributed architecture, the resources management can be performed by using different devices distributed among the city. There are many recent proposals for distributed resources management, including cloudlets, fog computing, and edge computing. As part of these, fog computing has emerged as a promising technology for resources management in the Internet of Things, by using the computational capabilities of the set of devices located at the edge [1, 130]. With this strategy, data do not have to be moved to a central (and far remote) data center (usually in cloud) and, as a consequence, the network traffic and latencies can be reduced, while increasing fault tolerance and security safety.

3.5 Data Management in Smart City

Most architectures designed with explicit data management schemes are centralized. This means that even though data is collected from different sources spread all over the city (such as sensors, surveillance cameras, third party applications, external databases, etc.), data is accessible from a centralized site, usually in the cloud. For instance, in [141] Gubbi et al. propose a cloud centric vision for interaction between private and public clouds, later extended in [142] to propose an information framework particularly tailored for Smart City management. As shown in Figure 3.2, the data flow is clearly specified, including four layers, namely Data Collection, Data Processing, Data Management, and Data Interpretation. However, note that applications and services obtain the data from a centralized cloud computing platform.

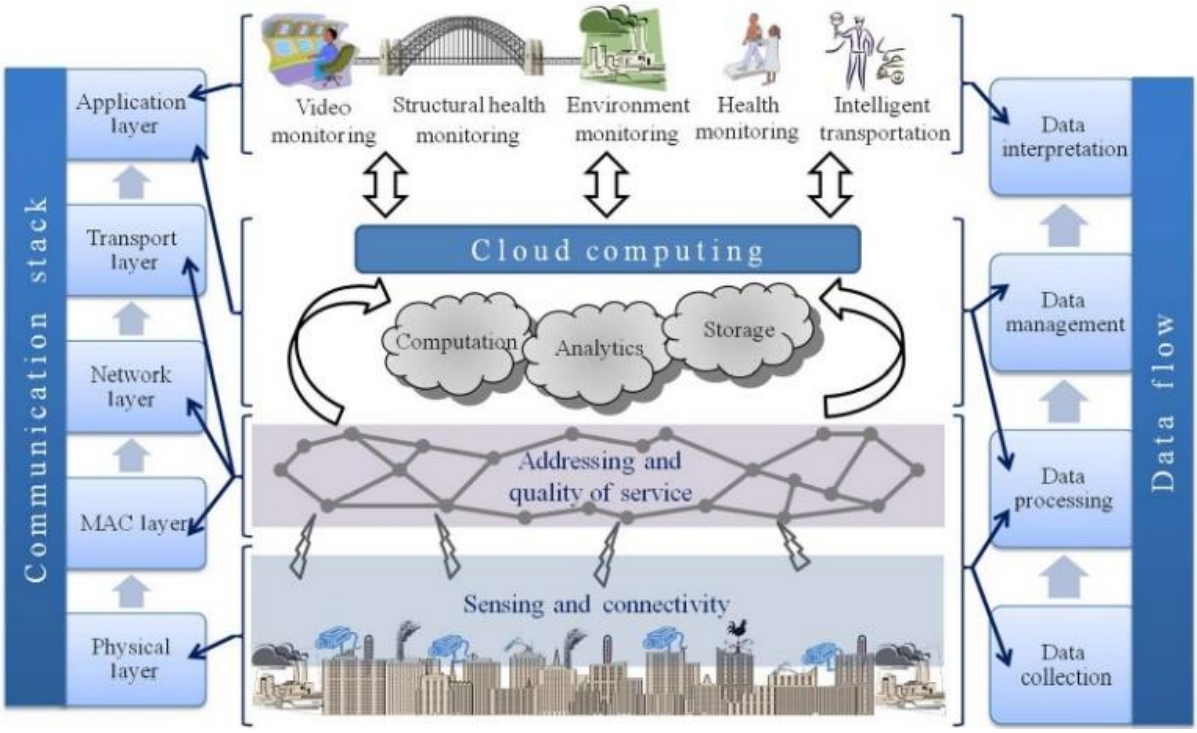


Figure 3.2 IoT architecture for Smart City[142]

In [143] Rathore et al. basically follow the same patterns but focus specifically on Big Data Analytics. This means that all collected data is preserved in the central cloud, and includes several additional data life cycle steps, such as data aggregation, data filtering, data classification, preprocessing, and decision making. In [144], Pena et al. also propose a Big Data centric framework for smart systems through Internet of Everything (IoE) but, basically, the model is similar to previous models in terms of data flow layers.

In [126], the author assessed that data management application acquires in the cloud. And then the author discusses about transactional data management and analytical data management. The transactional data management is related to topic of application database management which is organized somewhere very close to cloud environments. The analytical data management is part of applications demanding data storage to be utilized in business planning, problem solving, and decision support. Historical data along with data from multiple operational databases are all typically involved in the analysis.

Oppositely, few architectures propose a distributed schema for resource allocation and management, using technologies such as Fog Computing [130] or Fog to Cloud Computing [16]; however, none of them has an explicit focus on data management and organization. One exception may be found in [145], where Sarkar et al. explicitly address some issues related to data collection at fog level, and distributed temporal data storage also at fog level.

3.6 Barcelona Smart City

In this Section, we introduce a real Smart City scenario that will be used throughout this thesis to illustrate some implementation details of our architecture and evaluate the efficiency of this architecture in the Smart City scenario. Barcelona is located Northeast in Spain and is referred to as the second-largest city in Spain. Estimated population is 5,309,000 in 2016 which it makes the 6th most populous urban area in the European Union and the largest on the Mediterranean Sea [146]. In addition, Barcelona has almost 100 km2 area, allocating (average) 16,000 people per square kilometer (41,000/sq mi). Thus, the huge amount of people, in the wide geographical distance at Barcelona, makes it needed to deploy high level services and urban equipment to the city hall. According to an official report issued in 2016, the city hall prepared and installed some urban equipment for their citizens, such as 150,000 lamp spots, 80,000 parking slots, 40,000 garbage containers, and many other “smart” devices[147]. Recently, the city hall deployed more facilities aiming at meeting citizen needs and demands (innovative smart services and applications), setting Barcelona as a flagship city when discussing about the Smart City concept all over the world.

The Barcelona city designed an architecture to deal with city information sources (including IoT devices, sensors platform, etc.) along with Smart City applications, supported by a middle layer, referred to as City OS. Thus, the proposed architecture consists of three main layers, namely city information sources, City OS and Smart City applications, as shown in Figure 3.3[148]. The city information sources layer is covered with abundant data sources (including physical and non-physical sources) in the city and plays as a mediator (or broker)assisting on the data collection and sharing strategies required by the Smart City applications layer. The Smart City applications layer plays as a repository for a set of applications and services to be executed by the citizens.

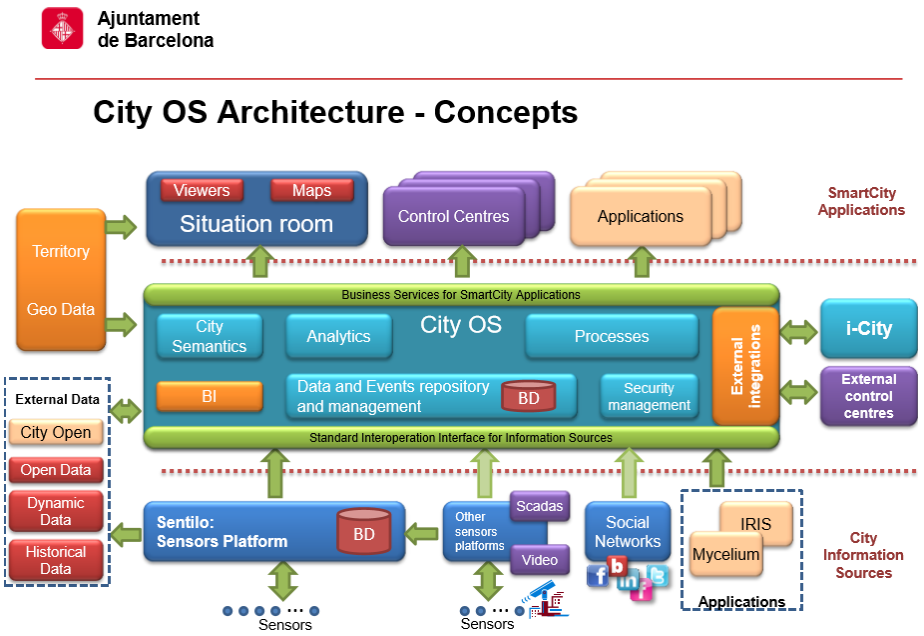


Figure 3.3 City OS architecture for Smart City in Barcelona [148]

More concretely, the City OS architecture includes four main stages to handle their tasks. Those stages are data repository, semantics, processes and ontology, as shown in Figure 3.4. The data repository stage aims to store all collected data from the city information sources. The processes stage provides a subset of processes for stored data. As it is shown in Figure 3.4, the process stage is able to read and write data to the data repository stage. Certainly, the processes stage can read the appropriate data from the data repository stage first, and then, the received data can run some specific processing, generating "sophisticated" data forwarded back to the data repository stage to be stored again. The ontology stage aims to provide a subset of ontology techniques intended to organize and analyze data for future usage. Finally, the semantics stage is responsible for applying some semantic techniques to the stored data thus making the data ready for the application/service layer.

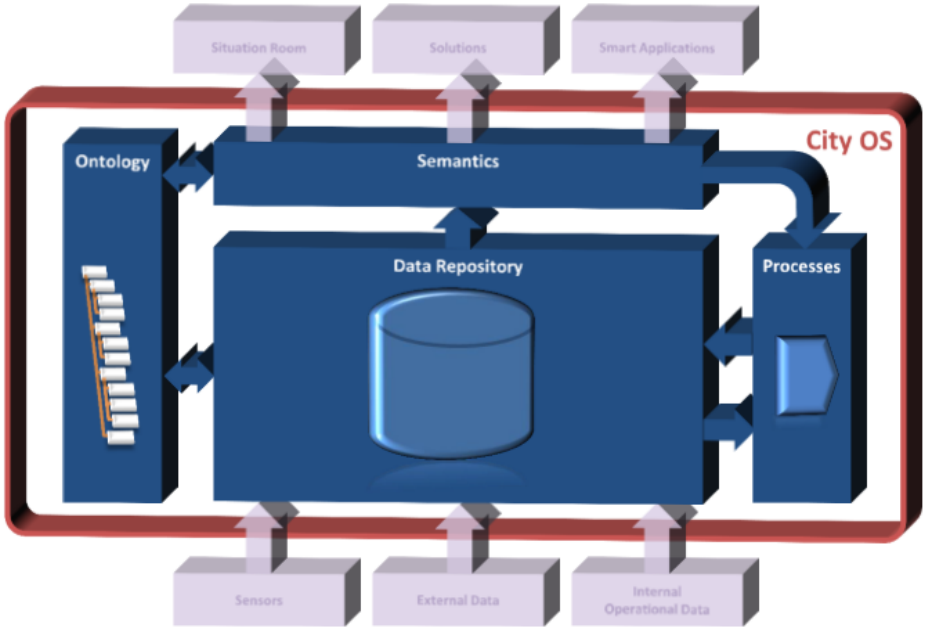


Figure 3.4 Layers of City OS architecture for Smart City in Barcelona [148]

From a physical perspective, see Figure 3.3, the city information sources layer includes a diverse variety of physical (e.g., sensors) and non-physical (e.g., social networks) sources. In particular, the Sentilo platform is part of the city information sources layer in the Smart City architecture proposed for the Barcelona city. The Sentilo platform is an open source software platform designed to visualize the data collected from the different sensors deployed in Barcelona, setting a network of sensors for sensing and collecting data in the Barcelona city over times [149]. The data collected from the sensors is archived and shared in the storage capacities built at Sentilo. Then, Sentilo may connect with the City OS to transfer these data for further objectives (including storing, processing or mixing the Sentilo data with another type of collected data/information) as shown in Figure 3.4.

In fact, the Sentilo platform has two main stages, data collection and data transmission. The data collection stage collects all possible data from the sensors deployed in the city, later transferred to the upper layer (which is City OS layer) through, see in Figures3.3 and 3.5. Therefore, Sentilo contains large repository capacities to save the historical sensors data. Nowadays, Sentilo goes beyond the preliminary Sentilo Cloud solution through its correlation with Waspmote sensors, Meshlium gateway and Libelium,aimed at building the first Smart Cities software platform which is based on experience and knowledge of the requirements of a large city such as Barcelona [150].

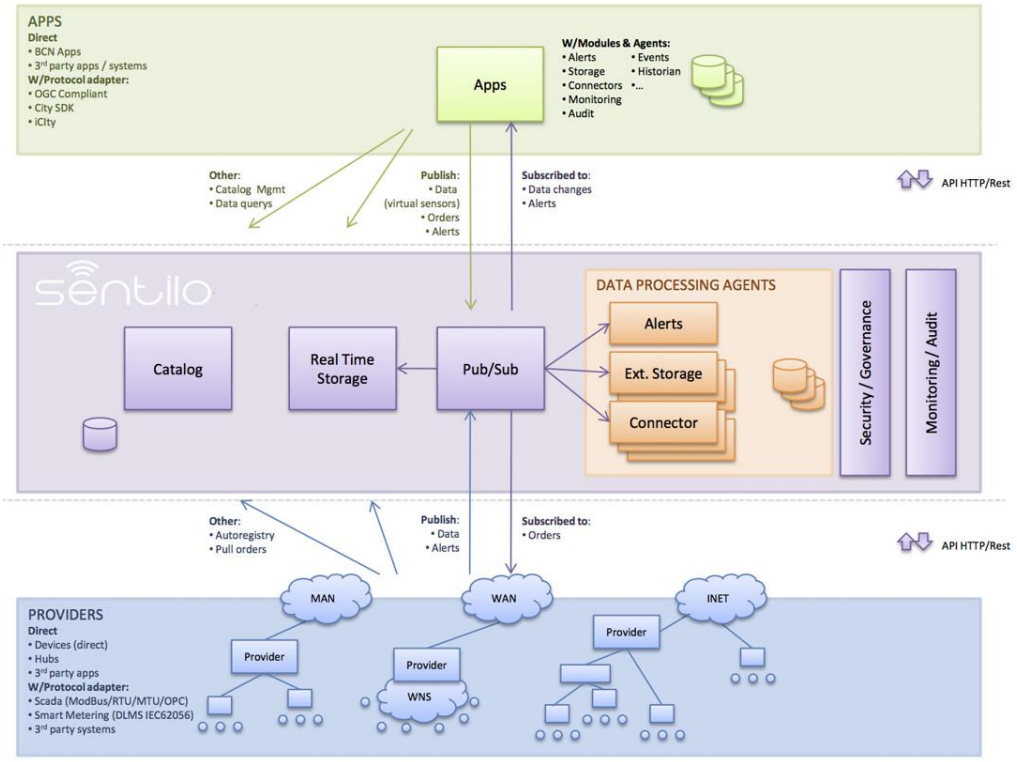


Figure 3.5 Sensors data architecture in Barcelona Smart City[151]

So far, we have shown that the Smart City architecture of Barcelona (as shown in Figure 3.2) proposed a centralized architecture for managing resources and data. In order to illustrate the real chances this model may have for a wide deployment we figure out some figures. First, we calculate the number of sensors data. Then, we will estimate the number of sensors data to come in a near future (in two main terms, full sensors coverage in the area of Barcelona and also extending the type of information to be collected from the sensors). After that, we show how we can optimize the volume of generated data –using some compression techniques–, through a distributed model. For instance, we show some data aggregation techniques to reduce the volume of data to be forwarded to the upper layer, seeking for an optimized data collection.

Chapter 4:

Data Management over a F2C Smart City Scenario

In previous chapters, we deeply explained how data can be managed in smart cities scenarios, considering cloud computing, turning into a centralized approach (i.e., Cloud Data Management, see[152]). We also discussed some research avenues driven by envisioned challenges in the area of data management Smart City scenarios. This chapter is focused on describing the main contribution of this thesis, i.e., the design of a data management architecture for a Fog-to-Cloud Smart City resources management structure.

To that end, we first design a comprehensive DLC model for smart cities, considering all data life stages. This initial task is divided into two subtasks:

- Designing a comprehensive scenario agnostic DLC (COSA-DLC) model.

  In Chapter 2, we ended up emphasizing the need for a novel comprehensive DLC model, to be easily mapped into any scenario and environment. Thus, this chapter works on that direction, proposing a comprehensive DLC model, showing the need for its easy adaption to any scenario – to keep it scenario agnostic–, as well as estimating the efficiency of the proposed model through the 6Vs challenges.

- Designing the Smart City comprehensive DLC (SCC-DLC) model, tailored to a Smart City context.

  As we know, data is an essential asset in any smart scenario, also referred to as "green oil" to emphasize the importance and relevance in such smart contexts. Focusing on a Smart City scenario, it is widely accepted that, in a general view, smart cities must deal with constraints coming up from the complexity linked to the data characteristics (including heterogeneity, high volume, etc.), thus requiring the need for an innovative model to manage these data. This model is inferred from the COSA-DLC, extended to meet the Smart City requirements. The SCC-DLC model is able to manage data in the Smart City to meet any demands of data stakeholders in the Smart City.

In a second step, we map the SCC-DLC model into the F2C Smart City. This task is also divided into two subtasks:

- Describing the F2C Smart City.

  The SCC-DLC model proposed in this thesis is an abstract model that can be easily implemented in a Smart City scenario. Before doing so, we first describe the structure of the city according to a F2C resources management architecture. The architecture is distributed and hierarchical, and organized into several layers, from the fog up to the cloud. This architecture will be used as a baseline for our data management architecture design.

- Proposing the F2C data management architecture.

  Data management in a Smart City is a complex task exacerbated by the large set of "components" building the city, such as an enormous amount of distinct IoT devices, computing models, network communication technologies, etc. Thus, all these aspects must be considered in our data management model (SCC-DLC) to make the solution work. F2C, yet in its infancy, comes up as a potential management framework bringing a solution for an efficient

resources management, especially for scenarios where diversity is a must. Thus, we leverage such a F2C solution as a baseline concept to enrich our data management architecture.

Indeed, this chapter is organized into five main sections. In Section 1, we describe our proposal for managing data complexity, i.e., the comprehensive, scenario agnostic, DLC (COSA-DLC) model. Plus, the comprehensive DLC model is evaluated with respect to the 6Vs challenges as benchmark test. Section 2 presents the SCC-DLC model (including main blocks and phases) with respect to the adaption of the COSA-DLC model. Plus, we report some benefits of the SCC-DLC model. Then, we describe our depicted Smart City scenario, defining the different layers in aSmart City scenario, and discussing the concept of F2C data management architecture. We show how F2C data management can be mapped into a Smart City scenario. Then, we highlight several advantages of the F2C data management vs the SCC-DLC architecture. Finally, we emphasize the main contribution in this chapter and include our related publications.

4.1 Proposing a COSA-DLC model

The comprehensive scenario agnostic DLC (COSA-DLC) model considers all phases of data management and organization, from data acquisition to data preservation and processing, and includes other fundamental aspects related to data quality and data security, among others. In addition, the COSA-DLC model can be easily customized and fitted to any scenario to guarantee the specific requirements while providing high level of data quality.

Some potential advantages of a COSA-DLC model are: i) managing and organizing global datasets for any future data discovery, integration, and processing; ii) providing easy customization and adoption to any science or scenario; iii) improving data quality levels in any specific context, and; iv) eliminating any additional waste and effort for designers, including data, software and system designers, to design their appropriate and efficient architecture.

The main organization of the COSA-DLC model is defined in three main blocks, named Data Acquisition, Data Processing, and Data Preservation. Each block, in turn, is further described into a set of more detailed phases, covering all cycles involved in the data life. In addition, each phase is specified in terms of the Data Lifecycle Management (DLM), which defines the phase's policies and actions, and the interrelation among phases.

4.1.1 Main blocks in the COSA-DLC model

The COSA-DLC model is defined as a modular three blocks structure (see Figure 4.1), with the Data Acquisition, the Data Processing and the Data Preservation blocks. These blocks are responsible for gathering, storing and organizing data for processing purposes, while guaranteeing high levels of data quality. This modularity eases the process of adaption to specific scenarios by tailoring this model to the specific scenario requirements.

The Data Acquisition block is responsible for collecting data into the system, gathering data from different sources, assessing data quality, and tagging data with any additional description required in the business model. Collected data can then be stored, through the Data Presentation

block, or processed, through the Data Processing block. The Data Processing block is responsible for performing the main big data processing, extracting knowledge or generating additional value, through complex data analysis techniques. The outcome of the processed data (higher value data) can be either delivered to end users, or stored for future additional data usage or reprocessing. The Data Preservation block is the responsible for data storage, performing any eventual action related to data curation. This data is ready for future publication or dissemination, or for further processing.

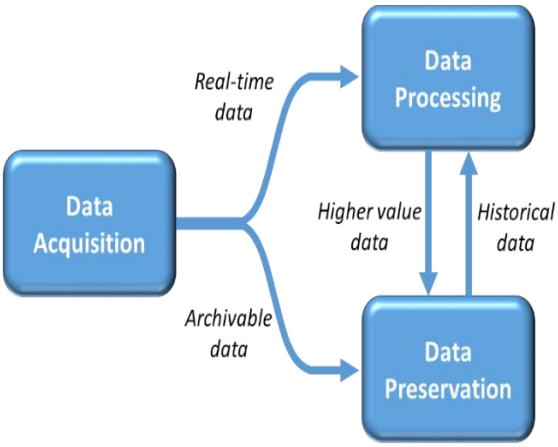


Figure 4.1The DLC model in blocks

The data flow is as follows. Data is initially created and collected through the Data Acquisition block. If data is immediately processed, this is considered real-time data; otherwise if it is preserved, this is considered archivable data. Note that either all or part of the processed data can also be preserved, and vice versa, i.e. these two data sets are not exclusive. When archived data in the Data Preservation block is used for processing, this is considered historical data. So the Data Processing block is able to use both real-time and historical data for processing. Finally, the data processing outcome can be stored back through the Data Preservation block: this data is considered higher value data.

4.1.2 Phases in each block for the COSA-DLC model

The proposed COSA-DLC model is organized into three main blocks, as described in the previous section. This set of blocks includes a sophisticated set of phases implementing all required tasks to make comprehension, agnosticism and adaption true. Thus, as shown in Figure 4.2, the Data Acquisition block is developed in four phases, the Data Processing block is developed in three phases, and the Data Preservation block is developed in four phases. The description of all functionalities and activities of each phase, together with the relationship among phases, is called the Data LifeCycle Management (DLM), and is presented next.

The Data Acquisition block is made by the following four phases, namely Data Collection, Data Filtering, Data Quality and Data Description:

- The Data Collection phase aims to collect data from all sources and devices, according to the business requirements and scientific demands. Specifically, it is responsible for:

  - Collecting data, directly and indirectly, from any valid source, such as basic or complex devices (sensors, smart devices), databases, web-generated data, third party applications, etc.
  - Managing the ranges of valid and trusted sources for data collection.
  - Exploring and discovering new sources for data collection.
- The Data Filtering phase is responsible for performing some basic data transformations in order to optimize the volume of data flowing from the collection to the quality phases. Particular data transformations are specific of the context and business requirements. However, filtering, aggregation, curation, sorting, classification, or compression, are some data transformations that could be considered as well.
- The Data Quality phase aims to appraise the quality level of the collected data. It is responsible for guaranteeing both, Quality Control (QC) and Quality Assurance (QA), in particular:
  - Checking the quality level of data and discarding or repairing low-quality data, according to the provided policies (QC).
  - Monitoring the quality of data flows and, in case of continuous failures, proceeding according to the provided policies (QA).
- The Data Description phase aims to tag data with some additional information for an optimal future usage. Any available metadata considered in the business model can be used, such as timing (creation, collection, modification, etc.), location or origin (city, country, coordinates), authoring, and so on.

  Once the data has been described appropriately, it can be used for either processing on real-time, or for archiving for future queries over historical data.

The Data Processing block consists of the following three phases, namely Data Process, Data Quality and Data Analysis:

- The Data Process phase provides a set of processes to transform (raw) data into more sophisticated data/information. These processes could include one or several internal steps, such as pre-processing or post-processing, depending on the particular business requirements.

  Data considered for processing can be either real-time –lately generated–, data (from the Data Acquisition block), or historical archived data (from the Data Preservation block). The output of this phase is considered higher value data, meaning that this data is more mature than the original (raw) input data.

- The Data Quality phase aims to appraise the quality level of processed data. It can check both QC to the output of the processing and QA to the processing procedure.

  This phase could seem redundant or repetitive with respect to the Data Quality phase in the Data Collection block; however, it runs specific checking targeted to the specific life cycles. In addition, in order to provide completeness and to guarantee a maximum level of quality, any additional quality appraising is always useful.

- The Data Analysis phase is responsible for developing all data analysis and data analytics for extracting knowledge and discovering new insights.

This phase is the last step in the procedure of value generation, and it is usually the natural interface with the end-user. Alternatively, this data can also be considered for storing, as part of the Data Preservation block, thus allowing future data re-processing.

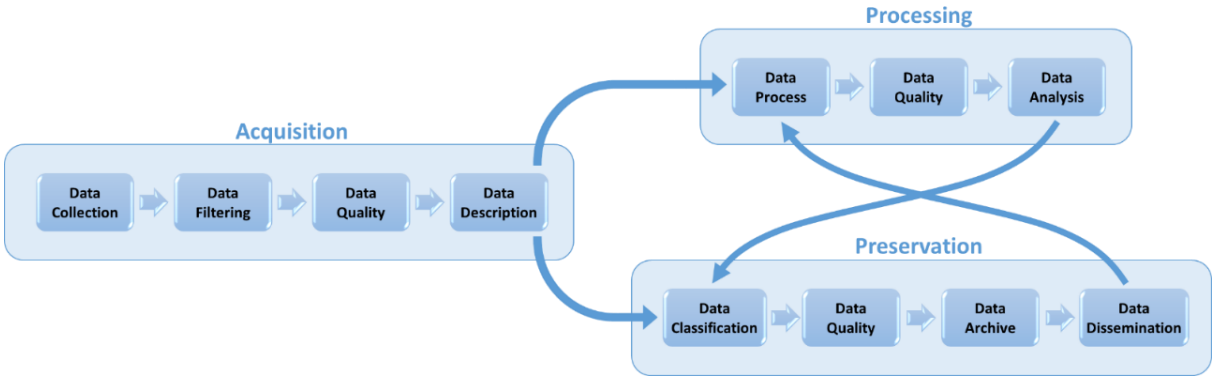


Figure 4.2 The proposed DLC model

The Data Preservation block consists of the following four phases, namely Data Classification, Data Quality, Data Archive and Data Dissemination:

- The Data Classification phase aims to organize and prepare data for efficient storage, by applying some optimization, such as classification, arrangement, compression, etc.

  Furthermore, some additional descriptive information could be also attached to this data related to the archiving policies, such as access permissions, privacy, expiry time, or sharing, use and reuse capabilities. In this phase, data provenance or data versioning could be considered.

- The Data Quality phase aims to appraise the quality level of classified data, before storing. It can check both QC to the output of the classification and QA to the preservation procedure.

  Again, note that this phase is specific for the data preservation block, and it is aimed at guaranteeing a maximum level of quality.

- The Data Archive phase aims to store a large set of high quality data in the available permanent or temporary storage resources. This phase must be able to perform long-term preservation over large amounts of data. It also takes over some additional tasks, such as data cleaning according to the corresponding expiry time or other business policies.

- The Data Dissemination phase aims to prepare archived data for private or public end-users' access. Any sharing procedures could be managed in this phase to guarantee access permissions, privacy, expiry time, or any other sharing capabilities.

This phase is the natural interface with the end-user for stored data. Additionally, this data can also be considered for processing, as part of the Data Processing block.

4.1.3 Comprehensive DLC Model Evaluation

As we mentioned in the previous section, several authors [27, 28, 41] previously highlighted a list of problems and challenges that should be considered in large and complex data management, often related to Big Data. These contributions have already identified several challenges, such as Value, Volume, Variety, Velocity, Variability, Veracity, and some others (note they all start with V). This set of challenges is known as the Vs challenges. Thus, the Vs challenges depict some strong barriers, difficulties and complexities for data management in Big Data related scenarios. However, existing contributions propose to work with different number of challenges, including 3Vs, 4Vs, 5Vs, 6Vs, or 7Vs challenges –perhaps more in the future–. In addition, we analyzed the appropriate challenges to be addressed in DLC models and considered the aforementioned 6Vs challenges. We also revisited most DLC models and evaluated them with respect to the 6Vs challenges as benchmark test. We concluded that although each model is adequate for its particular purpose, there is no any comprehensive model that addresses the 6Vs challenges completely. In this section we evaluate the proposed COSA-DLC model in order to demonstrate it is certainly comprehensive according to the 6Vs challenges.

- Value: The value challenge refers to the valuable information that can be extracted from (a huge volume of) data, after some processing and/or analysis steps.

The proposed COSA-DLC model has several merits to address the value challenge. Firstly, the Data Process and Data Analysis phases are precisely included for extracting value from data. Secondly, all Data Quality phases included in the model guarantee a high level of data quality and, therefore, data is more valuable. In fact, any DLC model, just because of its nature, induces to be designed for obtaining any kind of goal, or benefit. So this challenge can be assumed for any DLC model.

- Volume: The volume challenge refers to the huge volumes of data, in any format, that must be considered for management.

The proposed COSA-DLC model addresses this challenge in both the Data Collection phase – it is prepared to collect data from multiple sources–, and the Data Archive phase –, it should be designed to store large amounts of data. Again, any DLC model is able to address this challenge by definition.

- Variety: The variety challenge refers to the diverse types and formats of the data to be considered, mainly because data provide from different sources.

The proposed COSA-DLC model addresses this challenge in the Data Collection phase by collecting data, directly and indirectly, from any source, such as basic or complex devices (sensors, smart devices), databases, web-generated data, third party applications, etc. In this phase new sources for data collection are also explored, therefore expanding the candidate sources for data collection. In addition, other phases, such as Data Filtering, Data Description or Data Classification have been included precisely for supporting data organization and classification with high variety of formats.

- Velocity: The velocity challenge refers to the speed rate of data stream generation and the subsequent capability to process it efficiently. This challenge is closely related to performance.

The proposed COSA-DLC model has been designed for achieving high performance, both during data stream collection and during data processing. For this reason, a specific phase is proposed to manage each of these tasks, namely Data Collection and Data Process. Certainly, the final performance will depend on the particular resources deployment, but by considering specific phases, the design helps allocating specific resources in these steps.

- Variability: The variability challenge refers to the possibility that historical data varies its semantic meaning over time.

The proposed COSA-DLC model includes a Data Description phase to tag data for future usage and a Data Classification phase where additional tagging could be done, including expiring date. The Data Archive phase also offers the option to implement some data cleaning policies. Finally, in the Data Analysis phase, some data analytics processes could be implemented to analyze and predict eventual context changes.

- Veracity: The veracity challenge can also be understood from two perspectives, according to different authors' interpretations: data quality or data security. Data quality concepts include quality of control (QC) and quality of assurance (QA). And data security prevents datasets from any kind of modification from unsecured and unauthorized sources and devices.

The proposed COSA-DLC model includes a Data Quality phase in all blocks (Data Acquisition, Data Processing and Data Preservation), guaranteeing both QC and QA. First, all data is checked and if quality is too low (according to the business model), this can be discarded (QC). If a low quality level is reported continuously, the whole process can then be checked, in order to improve procedures for better quality and performance (QA).

Furthermore, the COSA-DLC model is able to address the data security challenge in different phases. Initially, by guaranteeing sources to be secure and trusted during data collection. Some additional metadata can also be included during the Data Description phase to implement some eventual encryption mechanisms. In addition, during Data Dissemination, different access policies can be defined and implemented to manage accesses, permissions, etc. And finally, a deep security analysis can be performed on data during the data quality phase.

4.1.4 Use Cases that illustrate the ease of adaptation of the COSA-DLC model

In this Section, we present two different use cases to illustrate how easy the customization and adaption of the proposed COSA-DLC model is to any kind of science or scenario. The first use case adapts the COSA-DLC model for data management in a Smart City scenario. The second use case adapts the COSA-DLC model into a library, which represents a scientific sample.

4.1.4.1 A DLC model for a Smart City

In the first example we use the Smart City of Barcelona as a use case to illustrate the COSA-DLC model adaption. Data in the Barcelona Smart City is currently managed through the Sentilo platform[153], a framework that collects data from different sources (mainly sensors, but also other information sources from the city), organizes and stores it, and provides a public interface to

access the datasets, either real-time or historical data. Figure 4.3 shows the Sentilo architecture, a middleware providing a unified access to the public data.

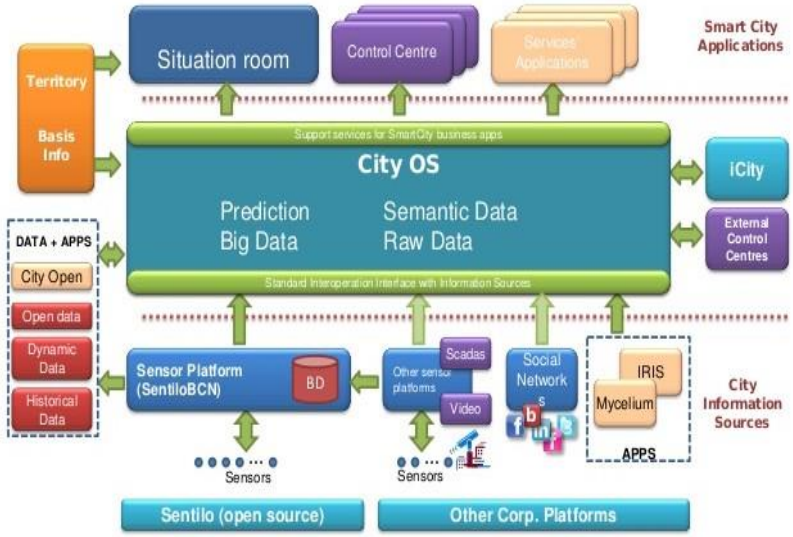


Figure 4.3 The Barcelona Smart City IT architecture

In order to adapt the COSA-DLC model to Sentilo, data acquisition, data processing and data preservation should be considered. Our proposal is illustrated in Figure 4.4. The Data Acquisition block includes Data Collection, Data Filtering and Data Description phases. Note that no quality control is performed in Sentilo as it provides the raw data as it gets it, only adding some additional descriptive data about timing and positioning. All collected data is archived in the Sentilo databases, and developers can access both real-time and historical data. The Data Preservation block includes Data Classification, Data Archive and Data Dissemination phases. The classification phase organizes information to be stored according to its type and format. And the stored data can then be retrieved through the interface managed in the data dissemination phase. Finally, some services are offered providing more processed data, although these services will be very customer dependent and then, no generalization can be easily made. For this reason, we include a Data Processing phase to allow some basic processing in the DLC model.

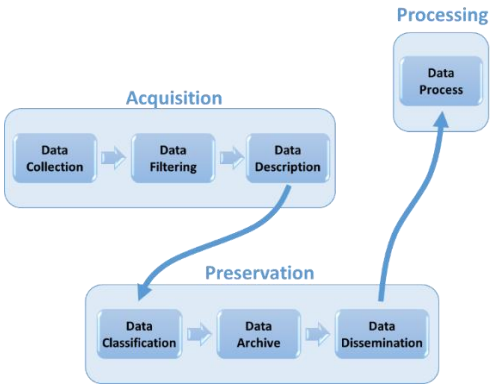


Figure 4.4COSA-DLC model proposal for Sentilo

4.1.4.2 A DLC model for a Scientific Library

In the second example we will use the Library of the UPC BarcelonaTech as a use case to illustrate the COSA-DLC model adaption in an eScience field. The library is connected to different university campuses to collect, aggregate and share all digital resources among internal libraries and departments. Several types and formats of information are collected by the library procedures, including books, journals, digital video, doctoral theses, examination records, etc., and all the information is available for online access, under registration and according to any eventual copyright statement[154]. The data in the UPC BarcelonaTech Library is currently managed through the framework shown in Figure 4.5.

The COSA-DLC model can easily adapt this framework by considering data acquisition and data preservation, as illustrated in Figure 4.6. The Data Acquisition block includes Data Collection, Data Filtering, and Data Description phases. Notice that in a Scientific Library many metadata is required to facilitate the catalog retrieval. The Data Preservation block includes Data Classification, Data Archive and Data Dissemination phases. In this case, some analysis is performed to check if different entries are referencing the same physical document in the quality phase. The dissemination phase provides a basic interface to handle advanced searching.

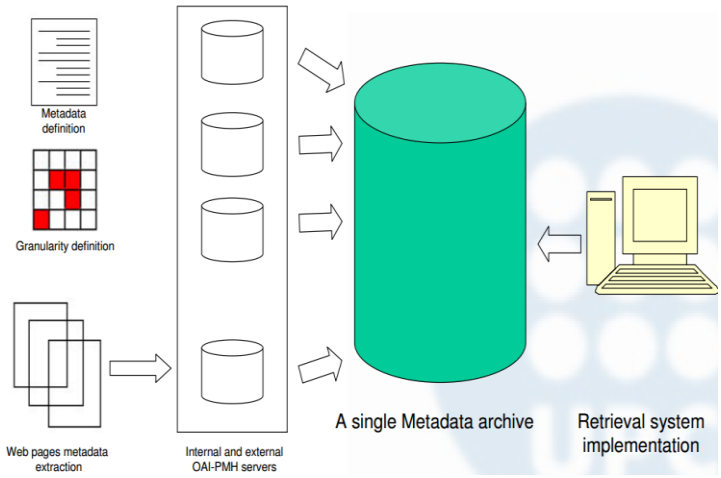


Figure 4.5The UPC BarcelonaTech Library architecture

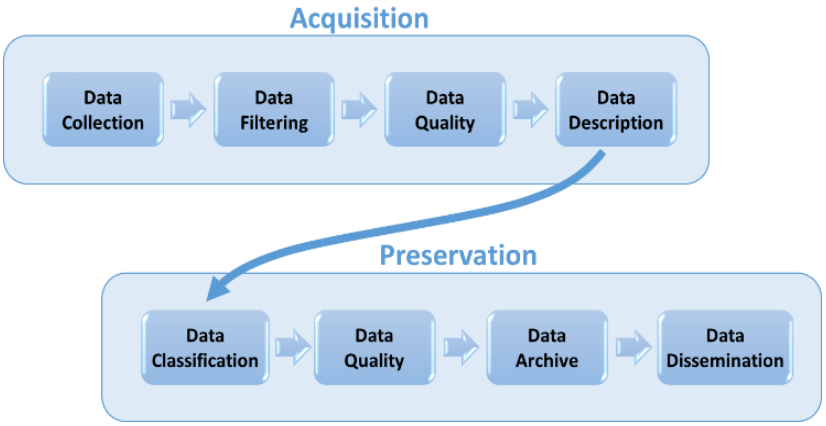


Figure 4.6COSA-DLC model proposal for UPC Library

These two use cases illustrate how easy the adaption of the COSA-DLC model into a variety of different scenarios and sciences may be. Indeed, adapting the model just requires selecting those phases that can be relevant according to the particular scenario requirements.

4.2 The COSA-DLC adaptation to smart cities: SCC-DLC model

The Comprehensive Scenario Agnostic Data LifeCycle (COSA-DLC) model has been proved to be: i) comprehensive, as it has been designed as an efficient and global data management model to address the set of 6Vs challenges for big data management (namely Value, Volume, Variety, Velocity, Variability and Veracity), and; ii) scenario agnostic, as it is easily adaptable to any scenario or scientific environment. In this section we extend the COSA-DLC model to a Smart City scenario, coining the Smart City Comprehensive DLC, or SCC-DLC model, for shorter [139]. The proposed SCC-DLC model has been designed for efficient data management and organization in the context of a Smart City.

In this section, we start by introducing the SCC-DLC model, showing how it can be inherited from the COSA-DLC model, and finally emphasizing its main benefits.

4.2.1 The SCC-DLC model

In Figure 4.7 we show our vision of the data life cycle in terms of main steps and data flow. We identify three major blocks, namely data acquisition, data processing, and data preservation. Data acquisition is one of the most important data related tasks in a Smart City, because the more information gathered from the city, the more complete and sophisticated services can be provided (as long as these data are verified and with high quality). So the data acquisition block is the responsible for collecting data, classifying data, assessing data quality, tagging data, applying any eventual data aggregation technique, and preparing the data for further usage. Data can then be processed or preserved. The data processing block is responsible for transforming data into information, knowledge, or any other higher value item, through complex analysis or analytical processes. This processed data can be either consumed by end users or stored for future usage. Finally, the data preservation block is the responsible for data archiving, storing high-quality data (curated in either the data acquisition or data processing blocks), and preparing the data for publication, or further processing phases.

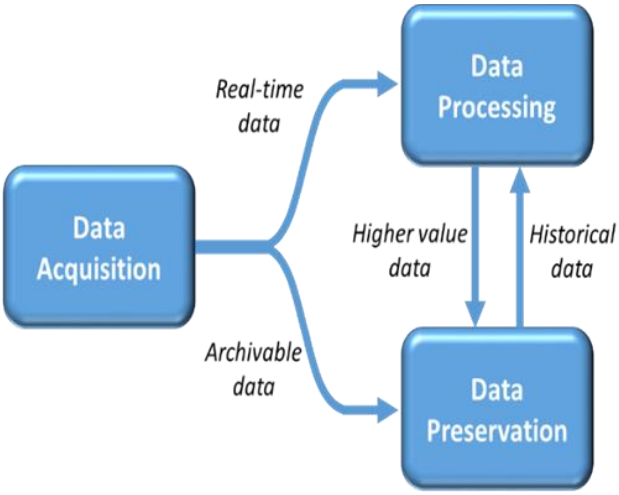


Figure 4.7 Data flow in the data lifecycle

The data flow is as follows. Data are gathered into the system through the Data Acquisition block, which collects data from different sources. If data are required for immediate processing, they can be considered real-time data; otherwise, if they are prepared for storage, they can be considered achievable data. Notice that either all or part of the processed data can also be preserved, and vice versa, i.e. these two data flows are not exclusive. When archived data, from the data preservation block, are used for processing, these are considered historical data. Hence, the data processing block can use both real-time and historical data for processing. Finally, the results of data processing can be consumed by end users or stored back through the data preservation block: in this case, these data are considered to be higher value data.

In Figure 4.8 we show the SCC-DLC model adapted to the smart cities scenario. As seen in the figure, each block is implemented through a set of phases to fulfil the required functionalities, as follows:

- The data acquisition block includes the data collection, data filtering (which performs some optimizations, such as data aggregation), data quality (aiming to appraise the quality level of collected data), and data description (tagging data with some additional information) phases.
- The data processing block encompasses the data process (which provides a set of processes to transform raw data into more sophisticated data/information), and data analysis (implementing some analysis or analytic approaches for extracting knowledge) phases.
- The data preservation block consists of four phases which are the data classification (aiming to organize and prepare data for efficient storage), data archive (storing data for short and long terms consumption), and data dissemination (publishing data for public or private access).

Notice that it is not necessary to implement any data quality phase in the data processing nor in the data preservation blocks because all data flowing to these blocks has previously been checked for quality in the data acquisition block. A complete description of each phase and its behavior can be found in below:

- The Data Acquisition block contains all phases defined in the original comprehensive COSA-DLC model. Their management is described as follows:

Data Collection, responsible for:

- Collecting data directly from physical devices spread along the city, such as sensors, surveillance cameras, users' smart phones and vehicles, and so on.
- Collecting data indirectly from other city sources, for instance, data created in city's local business or public institutions, and offered to the city as open data for smart services.
- Exploring and discovering new data sources that may extend the available data scopes at the city.

Data Filtering, responsible for:

- Applying some methods for data optimization, such as data filtering, data aggregation, data compression, data polishing, and so on. They are intended to optimize the volume of data managed in the system.
- Classifying or sorting data in order to provide enhanced performance. The actual classification will depend on the city's business model.

Data Quality, responsible for:

- Checking the data quality level (namely Quality Control) according to different techniques and algorithms. The particular quality methods required will depend on the city requirements.
- Discarding or repairing low quality data, according to the city's requirements and policies. In case of continuous failure, the data source could be blocked.
- Monitoring the quality of data flows and, in case of continuous failures, proceed according to the provided policies (namely Quality Assurance).

Data Description, responsible for:

- Tagging data with additional description for optimized future retrievals.
- Any metadata considered in the business model can be used, such as timing information (creation, collection, modification, etc.), location positioning (city, country, GPS coordinates), authoring, privacy, and so on.

- The Data Preservation block contains all phases defined in the original comprehensive COSA-DLC model except the Data Quality phase. The reason is that in the context of this Smart City, all stored data come from the Data Acquisition block and, therefore, its quality is granted. The phases' management is described as follows:

Data Classification, responsible for:

- Classifying and organizing data before storing, according to the city's business model.
- Adding some additional metadata regarding storage, such as expiry time, usage and reuse capabilities, security level, and so on.

- And eventually, implementing the corresponding management techniques in order to implement any data versioning, data lineage or data provenance.

Data Archive, responsible for:

- Storing (large sets of) data collected and processed in the city. Data will be stored in temporal sites, distributed along the city, and a selection of data (aggregated) will be permanently stored in the cloud.
- This phase is responsible for the long term preservation, but also responsible for some additional tasks, such as data cleaning according to the corresponding expiry time, or implementing other business related policies.

Data Dissemination, responsible for:

- Providing a user interface for safe private or public access to stored data, and managing data sharing according to the access permissions policies.
- Implementing the protection, privacy and security policies according to the business requirements.

- The Data Processing block contains all phases defined in the original comprehensive COSA-DLC model except the Data Quality phase. As with the Data Preservation block, the data quality checking is not necessary. Their management is described as follows:

Data Process, responsible for:

- Performing all data processing required in the application or service to convert raw data into some more sophisticated, higher level information, which provide smartness to the service. These processes could include one or several internal steps, such as pre-processing or post-processing, depending on the particular applications requirements.

Data Analysis, responsible for:

- Performing all deep data analysis and data analytics algorithms for extracting knowledge and discovering new insights. Again, the analysis or analytics processes tightly depend on the users' application or service.
- This phase also provides a user interface for accessing the results of data processing of an application or service. Alternatively, processed data can also be considered for archiving and stored.

Note that processed data can be either consumed directly by the end-user, or stored back to the system to allow data re-using and data re-processing.

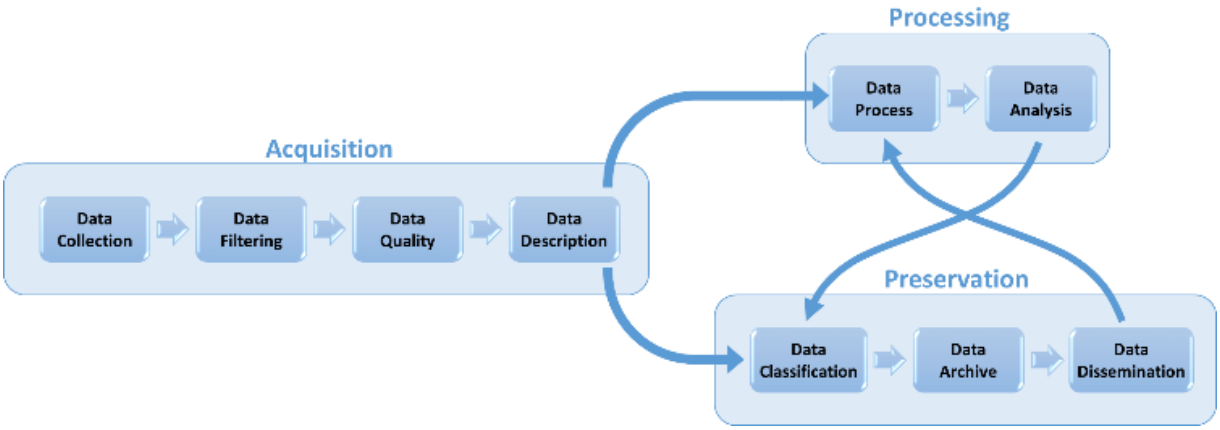


Figure 4.8 The SCC-DLC model

Finally, notice that all gathered data are accessible for smart cities' services consumption, usually through some sort of open access interfaces. In our proposal, we characterize data according to its age, ranging from real-time to historical data. For instance, real-time data is the one generated and just consumed, generally in critical very low latency applications. Such real-time data entails some implicit proximity constraints, because these data are difficult to be critical in remote services. Alternatively, data becomes historical (older data) as long as it is accumulated and stored on files or databases. In this case, historical data can be considered to be farther away (even if originally close) because accessing data from cloud, for instance, requires higher latency. We also consider that real-time critical data is requested in relatively small sizes, because very large volumes of data can hardly be considered for real-time. On the other hand, historical data can be requested in any, small or large data sets, and any type of fast or complex processing is expected to be done.

4.2.2 Advantages of the SCC-DLC model

We conclude that our SCC-DLC model has several advantages for data management in Smart City as show in below:

- Can be applied to any Smart Cities scenario easily
- Covers 6Vs challenges as a widely accepted concept of Big Data
- Provides a comprehensive model for data management in Smart Cities
- Gives some clues to developers and managers of Smart Cities to handle their enquiries regarding data life and stages (from creation to consumption)
- Organizing and managing data without any limitation about hardware and software
- Making facility to have standardization and globalization for the Data management model in the Smart Cities.
- The model considers data during their whole data life cycles, from production to consumption and cleaning, including storage and processing

4.3Scenario Description: the F2C Smart City

The Smart City is furnished with different type of sensors (such as temperature, electricity meter, and so on.), which are able to collect huge volumes of data in the city. The collected information will be then sent to computing devices (such as traffic light, smart phone, or something else) to do some initial processing and storage as shown in Figure 4.9. As you can see in this Figure, Edge-Data-Sources can be any type of sensor in the city. In addition, the Fog-Device can be any powerful device among Edge-Data-Sources as long as it supports Edge-Data-Sources coordination in the same location.

As shown in Figure 4.10, there are many different areas in the city, called Fog-Areas. A Fog-Area covers several Fog-Devices in the city. Similarly, a Fog-Leader node is defined as a node in the same area with capacity to organize and manage the resources in the Fog-Area.

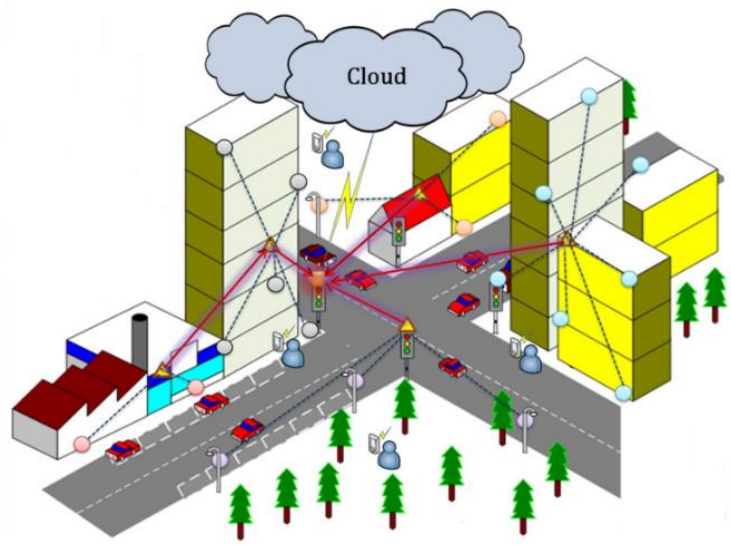


Figure 4.9 Edge-Data-Sources and Fog Device

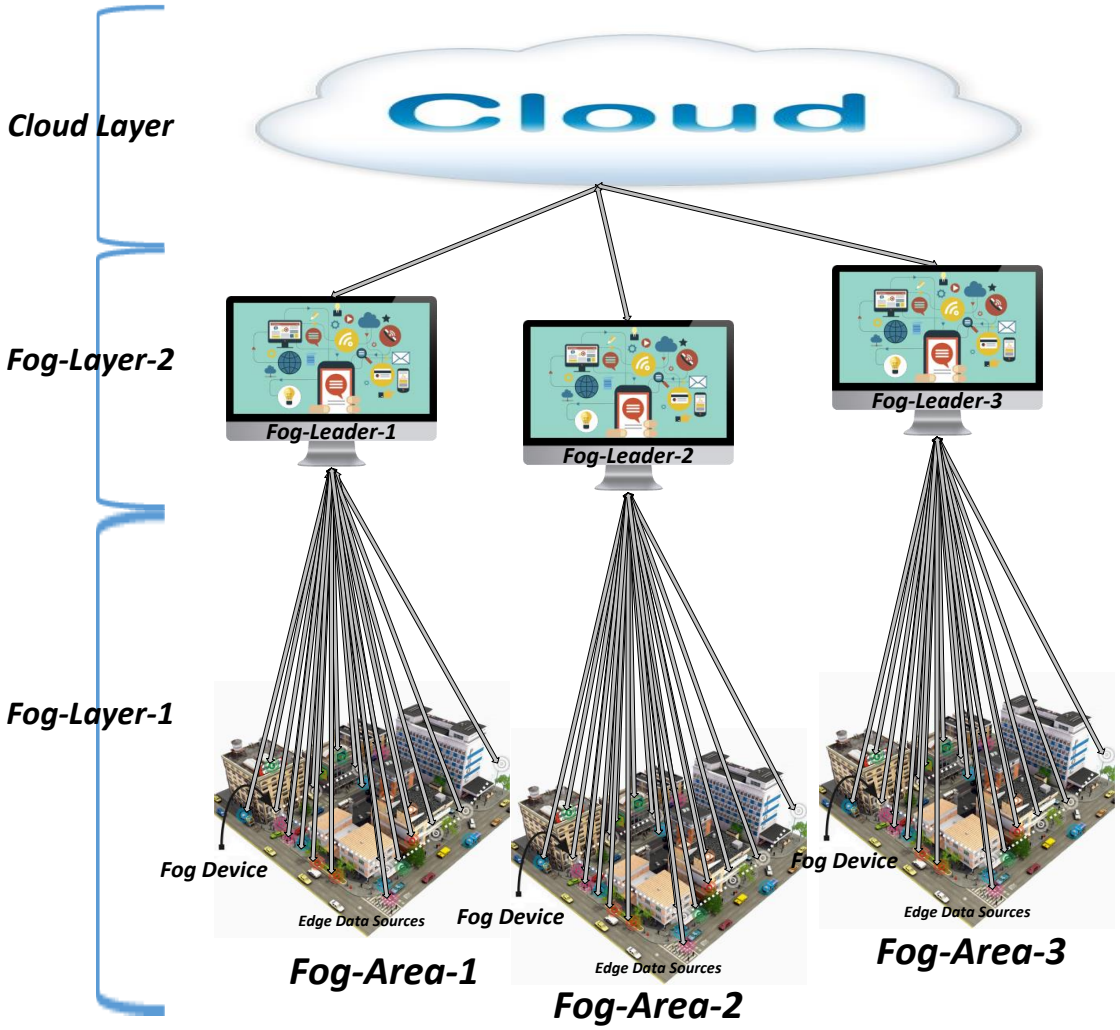


Figure 4.10 Our depicted Smart City scenario

As the number of Fog-Areas can be large in the wide city, a set of leaders will be coordinated by a higher level leader, becoming a hierarchical multi-layer structure, as shown in Figure 4.11. In addition, the number of layers will depend on the system size, urban city structure, business model, city manager organization, and etc. However, for simplicity, we will use a three layers architecture as shown in Figure 4.10. So, we depict our Smart City scenario with three different layers, namely Fog-Layer-1, Fog-Layer-2, and Cloud layer as described next.

- Fog-Layer-1 is the closest layer to the users in the city. This layer covers different Fog-Areas, Fog-Devices, and Edge-Data-Sources in our scenario as shown below.
  - Edge-Data-Sources: As shown in Figure 4.3, there are varieties of data sources in the city which include fixed (like sensors, camera and etc.) and portal devices (like smartphones, vehicular sensors and so on). These data sources are namely called edge-data-sources in our scenario.
  - Fog-Device: In our scenario, the edge-data-sources are managed and controlled by Fog-Devices. In addition, we consider that the strongest nodes among the edge-data-sources can be selected for the Fog-Device position in the city. In fact, a Fog-Device is supposed to have more capability for processing and storage than other edge-data-sources.

  - Fog-Area: In our scenario, the city can be divided into small size distance spots, we called “Fog-Area”. In addition, each Fog-Area covers different edge-data-sources and Fog-Devices.

- Fog-Layer-2 is an intermediate layer between cloud and Fog-Layer-1 layers. So, this layer is not close, like Fog-Layer-1, to the users and devices in the city, but it is not that far as cloud layer is. However, this layer is still located somewhere in the city. These layer covers with Fog-Leader as described in below.
  - Fog-Leader: In our scenario, we consider that Fog-Leader is positioned as strongest nodes among Fog-Devices. So, in fact, the Fog-Leader has more capacity in terms of processing and storage.
- Cloud layer is located at the top position of our scenario. Cloud covers the almost unlimited resources in terms of processing and storage. In addition, cloud layer is positioned in a place which can be very far away from the city (sometimes, in a different city or country or continent). So, the communication between city and cloud layer is a very important concept in this scenario. Currently, the new generation of cellular networks (like 4G or 5G) is used to build this network connection between cloud technologies and the Smart City scenario.

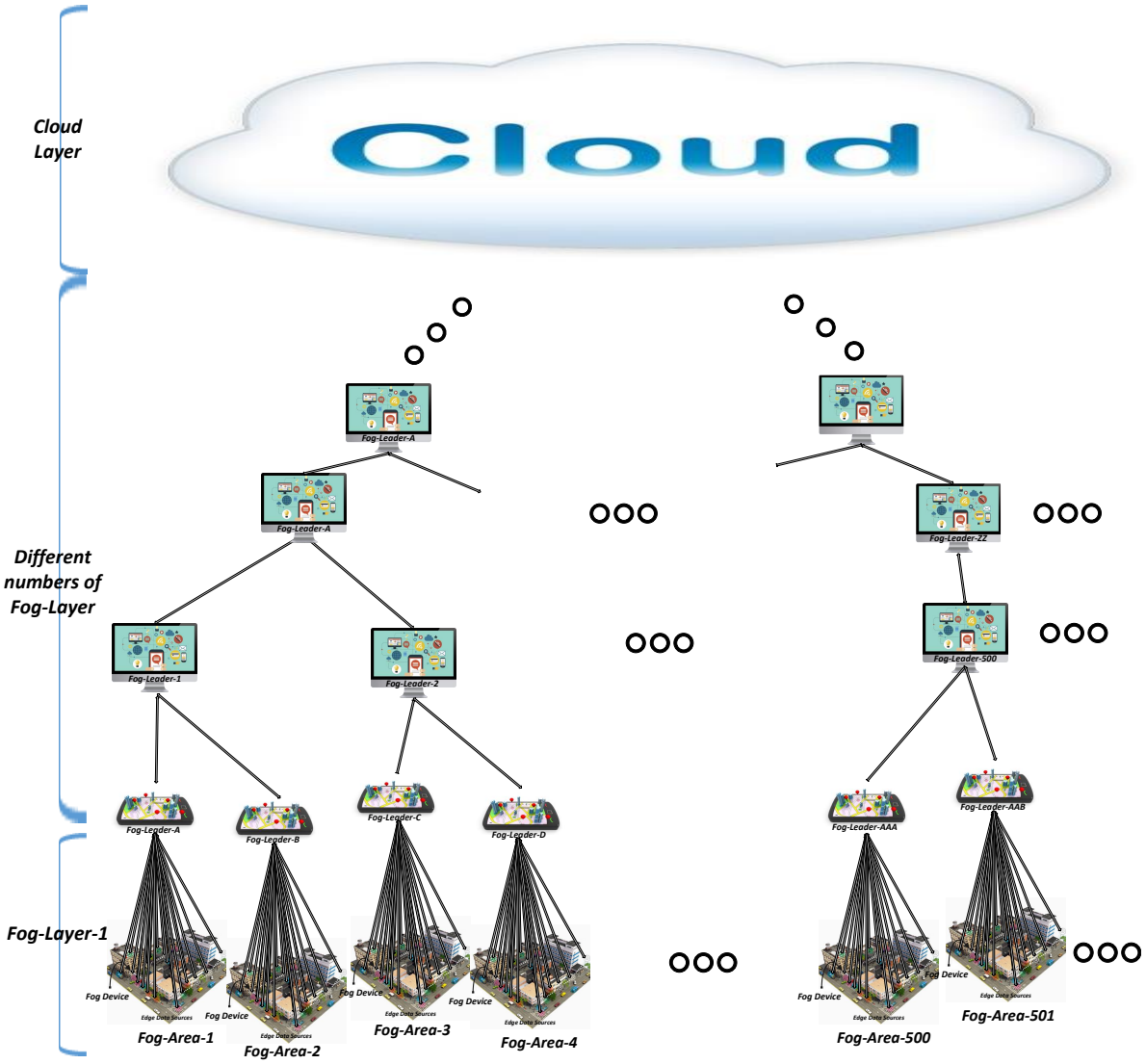


Figure 4.11Different numbers of Fog-Layers and Fog-Leader

4.4 The F2C Data management architecture

The distributed hierarchical F2C resources management architecture provides an interesting framework for data management in the context of smart cities, according to our SCC-DLC model proposal. In this section we present a novel architecture for efficient fog to cloud data management in smart cities, consisting on the mapping of the SCC-DLC model onto the Smart City F2C resources management architecture. Our model is illustrated in Figure 4.12. Notice that data acquisition is mainly performed at fog layer 1, as well as some basic data processing and data preservation actions. The fog layer 2 can enhance the data processing and data preservations capabilities of level 1 by providing higher computing capabilities. And finally, the cloud layer will be responsible for more complex and sophisticated data processing over a much broader set of (presumably historical) data, as well as the responsible for permanent data preservation.

In the following subsections the functionalities of each data lifecycle block in this architecture are described and, then, we discuss the advantages of our model.

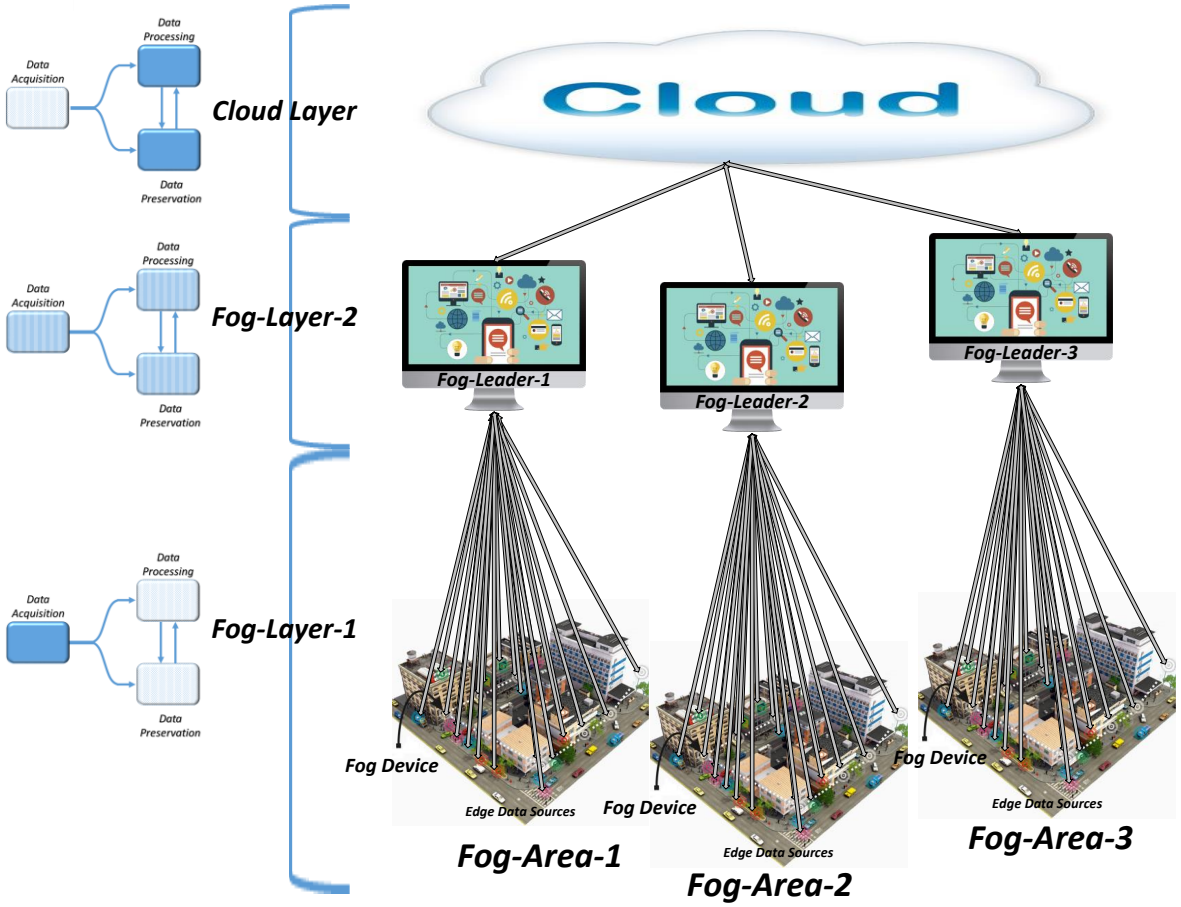


Figure 4.12 Mapping of the SCC-DLC model onto the F2C architecture

4.4.1 The Data Acquisition

Data acquisition is mainly performed at Fog layer 1. In fact, all sensor devices (such as the sensors network deployed throughout the city, but also surveillance cameras or sensor data from smart phones) are part of fog nodes at this level according to their respective location. Therefore, most data are collected at fog layer 1. There can eventually be some additional data collected from web services or third party applications, all collected at cloud level (where web services run), but these will be a small data set compared to the vast volumes of sensor generated data.

As long as the data are being collected, the following phases from the data acquisition block can also be performed at fog layer 1, where a reasonable amount of computing resources is available. For instance, the data filtering phase can apply filters to remove redundant data and can apply some data aggregation techniques to further reduce the amount of data to be managed. Data quality can also be implemented at this fog layer, assessing and guaranteeing higher data quality. And data description can be performed in order to tag data according to the city business model considered, for instance, timing information (creation, collection, modification, etc.), location positioning (city, country, GPS coordinates), authoring, privacy, and so on.

Data collected at fog layer 1 will be periodically moved upwards to layer 2, and data collected at layer 2 from a set of fog nodes at layer 1 will be combined and periodically moved upwards to the cloud level, which will collect the whole data set from the city. Notice that data at fog layer 1

can be immediately used at this same level (real-time data), so there is no need to urgently move these data to higher levels and, therefore, the frequency for the periodical upwards data movements can be strategically decided in order to accommodate it to the network traffic.

4.4.2 The Data Preservation

Data are generated at fog layer 1, but gradually moved upwards to the fog layer 2, and upwards to the cloud layer, where they will be permanently preserved. So, in fact, the F2C hierarchy acts as a reversed memory hierarchy, where data are created at the lowest cache level (fog layer 1) and moved upwards to main memory (cloud layer) instead of being created at main memory and moved to lower cache levels of the memory hierarchy.

Data generated at fog layer 1 will be temporarily stored at this level, allowing real-time applications an instant access to these data. The Smart City business model can decide the amount of temporal data that can be stored at this level, as well as the frequency of updating to upper levels. Similarly, data gathered at fog layer 2, consisting of data received from several fog nodes at layer 1, will be temporarily stored at this level 2. This will make up a set of less recent data (as it has been received after some period of time) but from a broader area, comprising the combination of the respective fog nodes' areas at layer 1. Finally, data will be permanently preserved at cloud layer, unless any expiry time is defined.

The different phases included in the data preservation block will be mainly executed at the cloud level, where the permanent storage is performed. Notice that these phases are not urgent and, therefore, their execution can be delayed to the time in which data reach to the cloud layer. This is the case of both the data classification phase –responsible for classifying and ordering data before storing, and eventually implementing the appropriate techniques for data versioning, data lineage or data provenance–, and the data dissemination phase –responsible for providing a user interface for public or private access to stored data, as well as for implementing any protection, privacy or security policies according to the city business requirements.

4.4.3 The Data Processing

Data processing can be performed at any F2C layer, according to the requirements of the application or service. For instance, critical real-time services will be executed at fog layer 1 in order to have a faster access to the (just generated) real-time data. Notice that accessing data locally inside the boundaries of a fog node is much faster than moving the data to a centralized cloud data center and, afterwards, reading these same data from the cloud to the local node.

Alternatively, deep computing complex applications will be executed at the cloud layer. Notice that: i) computing resources at cloud are unlimited and, ii) the data set of a high performance computing application will presumably be very large and, therefore, be part of the historical data set stored at the cloud layer. It is worth highlighting that in this case, when computation requires very high capabilities, adding more latency to the first access to data will not be significant in the overall performance.

Other applications will be executed at the lowest fog layer providing the required computing capabilities and the lowest fog layer containing the required data set. As a general rule, the closer the layer, the faster response times. An additional consideration in this case is when the required data is not present in the current fog node at layer 1, but can be accessed from either a node at a higher layer or a neighbor fog node at the same layer 1. This option may eventually be considered and solved using some sort of cost model to estimate the effects of both cases and proceed according to the lowest cost.

4.5Summary and contributions

In this Chapter, we have designed the Comprehensive Agnostic Data LifeCycle (COSA-DLC) model for any scenario, which is able to map with any science, scenario, and Big Data environment. Our COSA-DLC model comes with three main blocks, namely data acquisition, data preservation, and data processing, each describing a set of phases to properly manage the data. We have evaluated the COSA-DLC model with respect to the 6Vs challenges to obtain their efficiency in any Big Data environment. Plus, we showed two uses-case for depicting easy adaption of this model to any scenario and science (encompassing Smart City scenario and the scientific library science).

Next, we have proposed a comprehensive data management model for a Smart City scenario. For this purpose, we have shown how the COSA-DLC model can be easily mapped into a Smart City scenario, turning into the Smart City Comprehensive DLC (SCC-DLC) model. The SCC-DLC model has three main blocks and their related phases, namely Data Acquisition (including data collection, data filtering, data quality, and data description phases), Data Processing (consisting of data process and data analysis phases), and Data Preservation (encompassing with data classification, data storing, and data dissemination). The SCC-DLC model provides some desirable advantages to organize the most valuable assets in a Smart City, i.e., the data. Similarly, the SCC-DLC model is able to address successfully the 6Vs challenges.

And finally, we have designed the F2C data management architecture by tailoring the SCC-DLC model to the F2C hierarchical Smart City scenario. The F2C data management architecture proposes different layers, Fog-Layer-1 (including Edge-Data-Sources, Fog-Devices, and Fog Areas), Fog-Layer-2 (consisting with Fog-Leaders), and the cloud layer. We have shown that the data acquisition block is mainly handled at Fog-Layer-1 (with the high performance level), Fog-Layer-2 (with the medium performance level), and cloud layer (with the minimum performance level). Additionally, the Data Preservation and Data Processing blocks are performed at Fog-Layer-1 (with some basic data processing and data preservation actions), Fog-Layer-2 (with some more sophisticated data processing and data preservation actions), and cloud layer (with some advanced data processing and data preservation actions).

Therefore, the main contributions of this Chapter can be summarized as follows:

- Designing a comprehensive and scenario agnostic data management model (COSA-DLC) [155, 156], which has been proven to be comprehensive and easy to adapt to any scenario.

- Evaluating the COSA-DLC model with respect to the 6Vs challenges [155, 156], which proves the completeness of the model.
- Adapting the scenario agnostic COSA-DLC model into a specific scenario: Smart City. This adapted model is the Smart City comprehensive DLC (SCC-DLC) model, which keeps the feature to be comprehensive [138].
- Designing the F2C data management architecture by mapping the SCC-DLC model onto a F2C Smart City. This data management architecture is the core of the following contributions to be described in the following chapters. We have also shown the numerous advantages of this new data management architecture [138].

The novel COSA-DLC model provides some remarkable advantages and facilities as seen below:

- Organizing and managing data in any scenario, science, and Big Data environment through COSA-DLC model (without any limitation about hardware and software).
- Making facility to have standardization and globalization for the Data LifeCycle model.
- Giving opportunity to researchers in academia and industry to easily adopt our proposed COSA-DLC model to be applied to their scenarios with minimum efforts.
- Covering the 6 Vs challenges in any scenario and Big Data environment.

The main advantages of the F2C data management architecture design can be listed as follows:

- This architecture can benefit from the combined advantages of both, the cloud and the fog computing technologies, these are high computing and storage capabilities from the cloud layer and reduced network traffic and communication latencies from the fog layers.
- Real-time data accesses are much faster than in a centralized architecture. This higher speed is not only due to the reduced communication latencies of proximity, but due to the fact that accessing data from a centralized system requires the data to be moved first to the cloud, classified and stored there, and then moved back to the edge. So two times data transfer through the same path.
- By reducing the data transmission length, the security risks and the probability of communication failure are both reduced and, additionally, privacy can be easily enhanced.
- By having the just collected data available at fog layer 1, the network load is drastically reduced because some applications will be able to access these data locally, avoiding several remote data accesses through the network.
- By having the just collected data available at fog layer 1, the transmission to the cloud is not urgent and, therefore, it can be delayed without any performance loss. This allows additional optimization implementations, such as:

  - Performing some data aggregation techniques to reduce the volume of data to be transmitted upwards, without any computational constraint, as data do no need to be sent immediately.
  - Adjusting the frequency of the data transmission in order to use the network in periods when the traffic load is low.

- Traditional centralized systems define a low frequency policy for data collection from sensors in order to reduce the total amount of data to be transmitted in the network. By having the real-time data available at fog layer 1, the data collection frequency can be increased at this level without overloading the network and, therefore, providing more precision and accuracy from the sensed data at no additional cost.
- By defining a distributed storage hierarchy, data can be cached at different layers of the architecture and, for this reason, data access times can be easily reduced.
- In addition, the F2C architecture can manage data according to their initial location, which enables exploiting some locality features required in Smart City IoT contexts.
- From the processing and analytics point of view, this architecture allows a flexible interface in order to access the most convenient data for each service or application in an IoT context.
- During processing time, the architecture hierarchy provides an efficient structure that allows the application to access the nearest (and therefore fastest) data from the original data source.
- And finally, thanks to the distributed nature of the F2C data management model, it allows performing additional data related optimizations, such as providing high levels of quality, keeping high security and privacy standards, as well as reducing the global network performance.

The work done in this Chapter has conducted to the following publications:

- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, X.Yin, C.Wang, "A Data LifeCycle Model for Smart Cities", IEEE conference on ICTC 2016, Korea, October 2016.
- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, "Towards a Comprehensive Data LifeCycle model for Big Data Environments", IEEE/ACM BDCAT 2016, Shangai, China, December 2016.
- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, "A Comprehensive Scenario Agnostic Data LifeCycle Model for an Efficient Data Complexity Management", IEEE conference on eScience 2016, Baltimore, USA, October 2016.

Chapter 5:

The Data Acquisition Block

The Data Acquisition is the first and one of the most important blocks in any scenario because this is the block that provides the data (as the initial feed) for all upper layers from physical and non-physical devices. Several definitions of the data acquisition block in different scenarios may be found in the literature. In [157], the author argue that data acquisition senses the raw data generated by different physical and non-physical devices in a digital form for further storage and process. In addition, the produced data can be obtained with different types (such as image, document, sound and etc.) and formats (such as .jpg, .doc, .mp3 and etc.), turning into a huge volume of data in today's information technologies. Other authors tried to define some different sub-steps for the data acquisition block in specific scenarios. In [158]authors made three sub-steps for the data acquisition block in a Big Data scenario, that are data collection, data transmission, and pre-processing steps.

Some key differences come up when analyzing data acquisition block definitions from the academia and industry sides. First, some researchers support the fact that the data acquisition block is only responsible for gathering data from IoT devices, later sent to upper layers for further process and storage [152, 159]. In the particular scenario of smart cities, most contributions assume data will be collected in the city (as main task of the data acquisition block), and then upper layers/blocks are responsible for further data actions usually running at cloud [152, 159, 160]. Second, few authors mentioned that the data acquisition block (in particular data collection) can be included with some data actions to make data more sophisticated for further usage with other upper layers [157]. Those authors try to eliminate non-useful data from others data. So, they are demonstrating some kind of data actions (like cleaning, data filtering, and so on) to prepare data to be sent to upper layers for any further demands.

In our perspective, the data acquisition block goes far beyond a simple data collector from physical and non-physical sources (in particular for a smart city scenario). As we know, today´s IoT devices generate more than million data over times. However, some of these generated data are redundant or dirty (not useful at all in the future). So, we believe that the collected data must be refined with some available and basic technologies located at the edge of the network for the sake of data production proximity. In addition, due to the huge amount of data that is being constantly collected, this block should also be responsible for aggregating these data and, therefore, reduce the network traffic.

In the following section, we explain in detail (including definition, state of art, and challenges) the different phases envisioned in the Data Acquisition block (including data collection, data filtering, data quality, and data description). We show the Data Acquisition Block and its phases in F2C. In addition, we discuss how the phases (in data acquisition) can be handled in the F2C. Then, we propose the Data acquisition block in the city of Barcelona. Next, we describe how many sensors data (including current and future of the Sentilo platform) will be collected in the city of Barcelona. And then, we present some data optimization techniques (including data aggregation and data compression) through the data collection phase in Barcelona city. In addition, we highlight the efficiency rate of these techniques through the collected data in Barcelona city.

5.1 Phases in the Data Acquisition Block

We propose the data acquisition block in the SCC-DLC model to include four main phases, as shown in Figure 5.1. The proposed phases are “data collection”, “data filtering”, “data quality”, and “data description”. Next these phases are described.

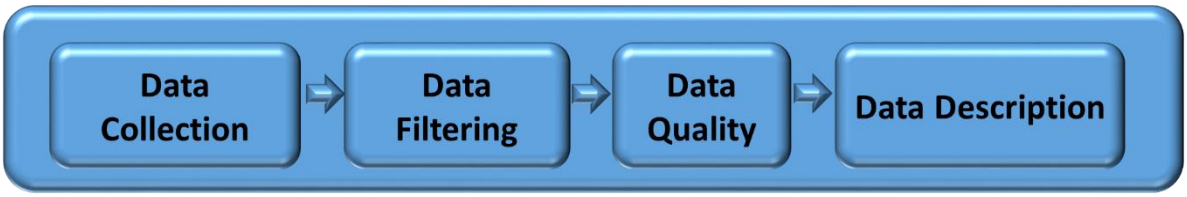


Figure 5.1 Phases in the Data Acquisition Block

The main objectives of the Data Acquisition Block are:

- Sensing and collecting all possible data from the smart city environment.
- Applying some data filtering techniques (such as data aggregation) to remove dark collected data.
- Appraising some quality techniques to control the quality of the collected data.
- Attaching some more information (including ownership, production time, and etc.) to the collected data for future usage in processing and/or storage.

5.1.1 The Data Collection phase

The Data Collection phase is the first phase in the Data Acquisition block, as shown in Figure 5.2.

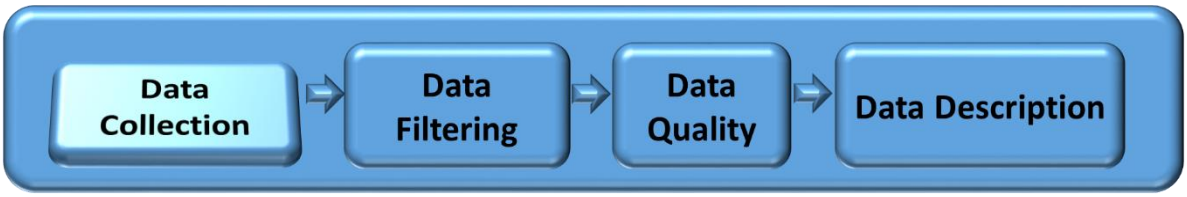


Figure 5.2The Data Collection phase in the Data Acquisition Block

5.1.1.1 Definition

Several definitions for this phase may be found in the literature, applied to different scenarios. In the area of the Internet of Things, the authors in [161]said that data collection is the ability of discovering and identifying “Things” and subsystems (statistic or mobile) for feeding the IoT data stores. In the area of Big Data, the authors in [158] mentioned that data collection techniques get help to grab raw data from a specific data generation environment. In a smart city scenario, data collection must gather all raw data generated in the city whose raw data come from installed

sensors or other non-physical sources [92, 152]. In brief, we can categorize related views about data collection definition as shown below:

- Available data: It means we stored data in different IoT data stores or a specific data generation environment. And now, we aim to collect our suitable data from this stored data pools.
- Non-available data: It means that we must sense and generate data from our environment by sensors, camera surveillance, and smart phones and so on.

5.1.1.2 State of the art

Collected data can be seen by different formats and types which maybe redundant and non-real-time data. We realized that all devices and sources can be categorized in a smart city scenario into the physical and non-physical devices:

- Physical devices: It means that there is a physical instrument which is installed somewhere in your environment to get data for further usage. In addition, the physical instrument can exist like fix (e.g. traffic light sensors) and portal devices (e.g. vehicular sensors) in the city. In the below categories, we show some types of the physical devices:
    - Sensors: Sensors can be seen in different types (such as acoustic, sound, vibration, automotive, chemical, electric current, weather, pressure, thermal, proximity, and etc.) to collect distinct numbers of information, as shown in Figure 5.3 [162]. Plus, we can sort the sensor networks to wired and wireless sensors [163]. It is very obvious that wired sensors must be connected by wire to somewhere. On the other hand, wireless sensors are able to connect to the next level by wireless communication technologies. In addition, energy and communication capabilities must be managed and optimized in this way of communication.

Figure 5.3 Different types of information and their sensors

- Human biometrics: Finger prints and signatures can be considered in this category. These physical devices are used to capture and save data for identity authentication and to track criminal for any future purposes [164].
- RFID (Radio Frequency Identification): RFID is a type of embedded communication. It consists of tags able to attach to any object for virtual identification. This tag also provides some space for electronically stored information which can exist by active, passive and battery-assisted passive types, as shown below [165, 166]:
  - Active type: This type is able to send their id signal periodically, which is considered as battery-powered type.
  - Passive type: This type is based on radio energy transmitted by the reader, which is not considered as battery-powered type.
  - Battery-assisted passive type: This type prepares a small on-board battery, which is able to transfer only in the presence of RFID reader.

- Non-physical devices: It means that data are saved directly or indirectly from physical devices somewhere (like data repositories) we must connect to in order to collect the data for our purposes. The main types of this non-physical devices can be existed in below:
  - Social networks: Day by day, users are more active to explore and interact their information and enquires in the Web such as LinkedIn, Facebook, Twitter, and so on. So, there are some techniques to catch user actives through Web and manipulate entered information with others as shown as below:
    - Log file: Log files are made by data sources to show their record activities in a specific file format for further subsequent analysis. For example, the web server log file formats have three main types (NSCA, W3C, and IIS) to depict user activities through websites [158]. Other examples can be given in "software-as-a-sensor" by physical sensors, stock ticks in financial applications, performance measurement in network monitoring, and traffic management.
    - Natural Language Processing (NLP): The user data must be understandable by machine. So, the NLP techniques are used to overcome this challenge. For example, NLP and NER (Named Entity Recognition systems are able to manage the emergency situations like car crash, earth quick and so on [92].
    - Web-Crawler: Web-Crawler is considered as general data collection applications for website-based applications such as web search engines and web caches [158]. The crawling process handle several policies like selection, revisiting, politeness, parallelization, etc.[163].

- Mobile Sensing and User Generated Data: There is a new way to generate data by users in smart devices (like smartphones) [162]. For instance, users can cooperate to produce data in smart city scenarios. In this example, there are some contributions showing how users can join each other to produce and share data in a smart city (such as Urbanopoly, Urbanmatch, csxPOI and etc.) [92]. Indeed, this way of data collection is highly important in these days because of low-cost timely sensing of the environments by users.
- Remote Database repositories: It means that data is stored in databases we can remotely connect to for taking our requirements [160]. In addition, there are varieties of available private and public data repositories in the city (such as companies, banks, hospitals, city hall and etc.).

So far we explained that data can come from different sources and highlighted the active open discussion among researchers intended to show how the useful knowledge may be extracted from the collected data. In [152], authors argue that all collected data must be sent directly to centralized computing model like cloud computing. However, other authors mentioned that data collection approaches must be covered with further actions and strategies, which is more than only data collection from physical and non-physical devices [161]. In addition, these researchers believed that data collection can impose some challenges and difficulties for network traffic, data storage, energy utilization and so on [152]. These actions and strategies can be listed as shown as below:

- Source Discovery Support: We believe that the data collection phase must be able to have some mechanisms for exploring new devices in your scenario. For example, there is some view in Internet of Things about source discovery system. In [161] mentioned that IoT devices and IoT applications are communicated with each other to announce which data is needed for services and then the services send their response about these needs. Indeed, those responses imposed devices to generate or update new data.
- Data Collection strategies: The data collection approaches can be followed by specific strategies in your scenario. For example, in [161]authors said that data collection from all “Things” can be considered as:
  - Temporal data collection: It means that data managers can define some rules for data collection approaches like specific interval times for data collections.
  - Modal data collection: It means that data collection can be controlled by some rules to match with data manager’s demands like some collecting data through specific elements.
- Mobility: In [161]authors emphasize that “things” of IoT devices can be moved everywhere per second. So, some solutions must be defined to enable IoT devices to access and transfer their information to IoT data stores. For example, a session-based synchronization system is proposed for mobile device to manage databases systems for running on mobile devices. Another example is Publisher/Subscribe-based systems is notification-based data for mobile devices.

5.1.1.3Objectives and challenges of an effective Data Collection phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data collection phase as part of a data management architecture. Some of these challenges, mainly those associated to the sensors and related technologies (for instance, reliability, accuracy, consumption, connectivity, etc.), are out of the scope of this research thesis. The main objectives and challenges listed here are these related to the resources organization and management. These are the following:

- Collecting data directly from physical devices spread along the city, such as sensors, surveillance cameras, users' smart phones and vehicles, and so on. This objective is highlighted several challenges as shown in below:
  - Data Volume: There is a huge number of data which are generated constantly by a huge network of physical devices.
  - Data Variety: Such huge number of physical devices generate data of different types, with different formats, accuracies, and so on.
  - Data Velocity: The massive data are being produced very fast and at more or less constant rates.
- Collecting data indirectly from other city sources, for instance, data created in city's local business or public institutions, and offered to the city as open data for smart services. Similarly, this data faced same challenges as shown in the previous objective (including data volume, data variety, and data velocity).
- Data generation (and therefore collection) must be frequency adjusted to the requirements of the city business model, and this has to be managed effectively.
- The data management architecture must provide fast access to real-time data for critical or real-time applications.
- The data management architecture must provide efficient mechanisms for accessing large volumes of historical data for computational intensive applications.
- The data management architecture must be able to manage data resources from mobile devices, thus managing mobility and all related issues, such as dynamicity, volatility, and so on.
- The data management architecture must be able to explore and discover new data sources that may easily extend the available data scopes of the city.
- Data are produced spread across city districts, so it seems an appropriate strategy to manage and organize the produced data through a distributed environment (distributed data management).

Our distributed hierarchical F2C data management architecture can effectively include and address most of the challenges listed above. This will be described and discussed in subsection 5.2.

5.1.2 The Data Filtering phase

The Data Filtering phase is located after the Data Collection phase in the Data Acquisition block as shown in Figure 5.4. This phase will be described in detail in the below subsections.

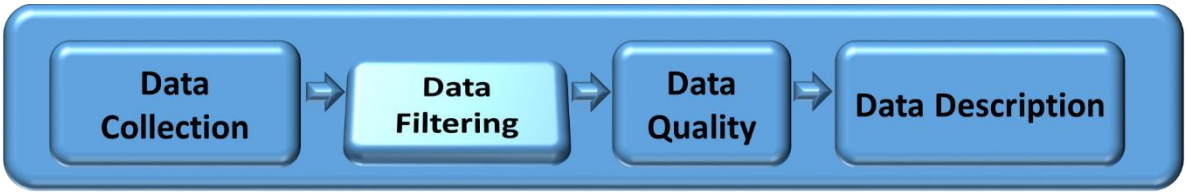


Figure 5.4 The Data Filtering phase in the Data Acquisition Block

5.1.2.1 Definition

The data filtering phase plays an important role in today's data world because data will be produced by many devices from different environments and scenarios every moment. So, these data can be redundant and useless for future access and storage. In [167], authors defined that the main objective of data selection and filtering logic is to provide the stability of their potentially huge data-handling systems and the usefulness of their Big Data.

In point of our view, we believe that Data Filtering phase is responsible for performing some basic data transformations in order to optimize the volume of data flowing from the collection to the quality phases. Particular data transformations are specific of the context and business requirements. However, filtering, aggregation, curation, sorting, classification, or compression, are some data transformations that could be considered as well.

5.1.2.2 State of the art

There are many contributions related to data filtering techniques. In Big Data and M2M scenario, [167] mentioned that data filtering platforms can categorized into general purpose filtering methods, Quality-of-Information (QoI) assessment techniques, and filter based on data classification. In addition, the last two categories (QoI assessment techniques, and filter based on data classification),are considered as "look into" data methods because general methods (like general purpose filtering methods) have difficulties to synthesize with semantics of data. On the other hand, there are vertical (bottom-to-top) and horizontal approaches through data filtering concepts. Currently, horizontal solutions are recommend to be used by M2M area networks, because horizontal solutions can be controlled by an API easy to configure for extending functionalities and layers, as shown in Figure 5.5 [167].

- General purpose filtering methods: This approach is normally considered as general data filtering methods which is mostly used for RFID data to handle some aggregation and filtering techniques. Basically, this approach follows some initial algorithms for performing removal and/or aggregation of duplicate data, erroneous data, outlier, and so on. Regarding some argument in M2M area networks, these algorithms are normally utilized for horizontal solutions and can organize fault tolerance in the network. However, there are some challenges about handling semantics of the data and the significant amount of data by these algorithms.
- Quality-of-Information (QoI) assessment techniques: This approach is considered as "look into" methods. This techniques focus on the importance of the contained data by deploying some evaluation techniques based on QoI scores. These algorithms are highly recommended for vertical solutions in M2M area networks.
- Filter based on data classification: This approach also belongs to the "look into" methods. This technique works with category of information rather than QoI scores. For example, Kobe solution used some standard of Machine learning to make classification of sensors data (including sound recordings, images and etc.) for further filtering purposes.

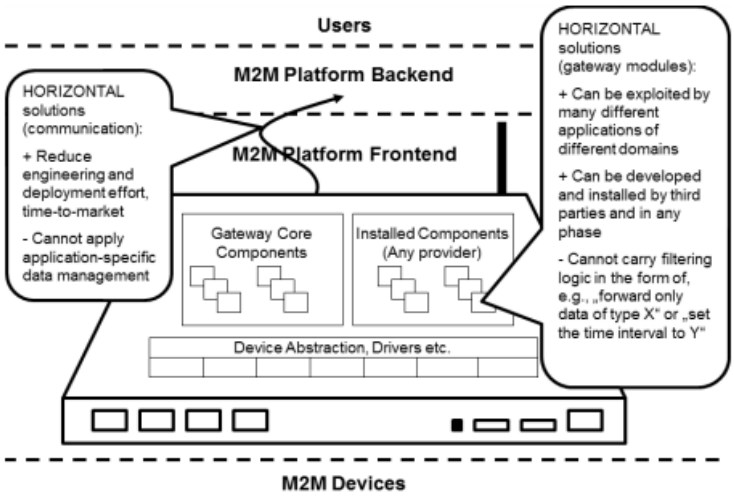


Figure 5.5 Horizontal and Vertical filtering solutions [167]

In [166], the authors said that there is system on a chip (SoC) in sensors to provide preprocessing abilities (some initial data filtering) for Wireless Sensors Network (WSN). Besides that authors explained something about data filtering in Smart Cities environments where they supposed that data filtering (including filters the unnecessary metadata and/or repeated data are also discarded [21]) is organized in cloud computing environment [159].

As mentioned above, data aggregation is a famous technique to handle data filtering and optimization in different scenarios. In fact, data aggregation provides a splendid facility as part of data management to do some kind of processing for gathering, reducing, mixing, or presenting information somehow as a summary [168]. The main objective of data aggregation techniques is reducing the amount of managed data, and can be obtained through diverse techniques, such as

data combination, data redundancy elimination, data compression, bandwidth reduction or power consumption reduction, just to name a few.

Recently, data aggregation has been tailored with the concepts of data and information mining progression, business demands and human analysis techniques, where data must be explored, collected, and presented in a report-based and shortened format in their networks [169]. There are some different views to do data aggregation in theoretical and practical scenarios. Traditional views concentrated to specific network devices and resources such as Wireless Sensor Networks (WSN) to manage data aggregation approaches [170-172]. The other view extends the previous view to go beyond ubiquitous and distributed scenarios (instead of focusing on specific devices and network) such as big data [169], cloud and distributed computing [169, 173], web technologies [174, 175], or real-time systems [172] as shown more details in the following sentences.

First, in WSN environments, sensors are located closer to the regions of the measured phenomena. So, it is very obvious the data aggregation techniques and approaches provide some help in such environments to perform data redundancy elimination, delay reduction, data accuracy (data quality), data security (reliability), traffic management, network scalability and minimizing overhead (bandwidth usage, processing requirements and power and energy wastage) [169, 171, 176]. In [177], the authors propose more sophisticated aggregation algorithms by proposing some soft computing techniques based on artificial neural networks, genetic algorithms, fuzzy logic models, and particle swarm techniques.

Second, in cloud computing environments, cloud computing provides (almost) unlimited, scalable as well as elastic resources. For this reason, cloud computing adopts some data aggregation approaches and techniques to produce high level and sophisticated final outcome. In [143], the authors provide a full data model from sensors nodes to cloud computing environments for a smart city scenario. This model has two main layers which are sensors nodes and cloud computing layers. The sensors nodes collect data from city and pass to the cloud computing layer. The cloud layer is responsible to data collection and aggregation, data filtering (including classification), and data processing (including preprocessing, processing, and decision making).

Indeed, with respect to distributed data aggregation, a recent survey [173] presents a taxonomy for distributed data aggregation approaches. They propose two main taxonomies, named communication and computation. The communication taxonomy focuses on the communication aspects (including communication/routing strategy and network topology). It is divided into structured (including hierarchical and ring protocols), unstructured (including flooding/broadcast, random walk, and gossip routing protocols) and hybrid data aggregation approaches. Alternatively, the computation taxonomy encompasses to decomposable functions (including hierarchic, averaging, and sketches basis and principles methods), complex functions (including digests basis and principles methods) and counting (including deterministic and randomized basis and principles methods) data aggregation approaches.

5.1.2.3Objectives and challenges of an effective Data Filtering phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data filtering phase as part of a data management architecture. The main objectives and challenges are the following:

- Several data cleaning or polishing techniques can be applied to the collected data in order to eliminate waste, duplication or other useless data. This processes provide higher data quality and contributes in better efficiency.
- Several data aggregation techniques can be applied to the clean data in order to reduce the data volume and, therefore, reduce data traffic and storage requirements through the system network.
- Several data compression techniques can be applied to the aggregated data in order to further reducing the data volume and, therefore, reduce drastically the system data set sizes.
- These objectives contribute in simplifying the data challenges of :
    - Data Volume, because applying the data filtering techniques drastically reduces the data size.
    - Data Variety, because the clean data helps organizing the data.
    - Data Value, because an increase in data quality impacts in its value.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. It is out of the scope of this research thesis proposing new filtering methods. We just show how easy and efficient is applying existing algorithms to this architecture. This will be described and discussed in subsection 5.2.

5.1.3 The Data Quality phase

The Data Quality phase is third phase in Data Acquisition block as shown in Figure 5.6. In addition, the data that meets the required quality standards continues to the Data Description phase. The more details of the data quality phase will be described in the below subsections.

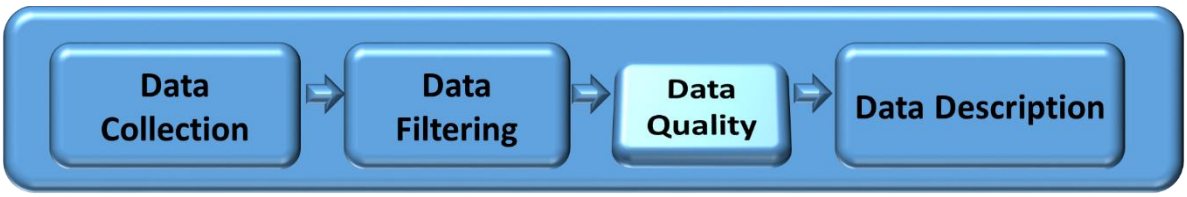


Figure 5.6 The Data Filtering phase in the Data Acquisition Block

5.1.3.1 Definition

In general term, data quality literature clearly show non-agreement on a single definition but mainly data quality problems imposed by incorrect data entry, missing information or other invalid data [23]. In fact, if data value can be considered as good level, it will be saved in your

storage media. Otherwise, the data must be discard or changed [178]. Now, we are listed some of definitions of data quality which are referred by [179, 180]:

- "Data Quality refers how well data meet the requirements of data consumers".
- "The currency of the data. That is the most recent time when it was updated".
- "The punctuality of the data item with respect to the application context".
- In IoT perspective, data quality refers to how collected data is suitable for providing ubiquitous services to meet IoT user's demands

"Data Quality Dimension" is a set of data quality [14]. Adaptability, Availability, Compatibility, Flexibility and Reputation are some examples of data quality. In general, we go beyond several works [14, 47, 181-183] to extract some of most important data quality dimensions. As you can see in Table 5.1, accessibility, completeness, interpretability, relevance, timeliness, and understandability are the most important data quality dimensions in our review.

Table 5.1 The most considered Data Quality Dimensions

Dimensions \ Reference	[14]		[182]	[181]	[183]	[47]	Total (1 <= X <= 6)
	General	Result					
(Access) Security							4
Accessibility							5
Accuracy							4
(Appropriate) amount (of data)							4
Believability							4
Clarity							1
Comparability							1
Completeness							6
Concise representation							2
Conciseness							1
Consistency							1
Consistent representation							4
Content							1
Cost-effectiveness							1
Currency							2
Ease of manipulation							1
Ease of operation							2
Efficiency							1
Flexibility							2
Format							1
Free of error							2
Freedom from bias							1
Importance							1
Informativeness							1
Interpretability							6
Level of detail							1
Objectivity							4
Precision							2
Quantitativeness							1
Relevanc(e/y)							6
Reliability							2
Representational consistency							1
Reputation							4
Scope							1

Sufficiency							1
Timeliness							6
Traceability							1
Understandability							5
Usableness							1
Usefulness							1
Value added							3
Variety of Data Sources							1

5.1.3.2 State of the art

There are several works showing how data quality and data quality dimensioned can be estimated in different scenarios/environments as shown as below:

- In general view, there are three common ways to find Data Quality dimensions (Intuitive, Theoretical and Empirical approach) as shown more details in below and Table 5.6 [14, 181]:
  - The intuitive approach is taken when the selection of Data Quality attributes for any particular study is based on the researchers' experience or intuitive understanding about what attributes are important.
  - The theoretical approach to Data Quality focuses on how data may become deficient during the data manufacturing process.
  - The empirical approach follows past experimental results

Table 5.6 A comparison of Data Quality approaches

Approach	Advantages	Disadvantages
Intuitive	Selected most relevant attributes	Do not see the voice of consumer
Theoretical	Provide a comprehensive set of data quality	
Empirical	Concentrate the voice of consumer	The results cannot be proven via fundamental principles

- In general [178], the author mentioned that modern data quality approach is trying to design a specific mechanism for making connection between customers and suppliers. This connection aims to provide matching between customers' demands and the suppliers' products.
- In [178], author propose a model to enhance quality for data to applications. There are three main activities and techniques to increase data quality as shown as below:
  - Check points: The main objective of this activity is to check quality of the value obtained. Check points is allocated in a quality failures place to prevent reworking of the preceding activities.

- Feedback loops: This activity provide connection between data customers, applications and data customers to improve quality level of data after sending and receiving their feedback
- Data destruction activities: This activity give helps to model for deleting deficient data in the system. So, if data level is not good enough and data cannot be repaired to meet the requirements, data must be discarded in this system.

- In Big Data environments, the author mentioned that there is different level of collected data in terms of quality. Plus, distinct types of data analysis techniques and applications/services may request for different level of data qualities. So, the author introduced some proposed data preprocessing techniques to improve data quality in big data systems. Typical data preprocessing techniques can be listed in below [163]:
  - Integration: The data integration techniques provide such kind of mixing techniques for data residing in different sources. Traditionally, data integration approaches can be considered in traditional database as shown in two below main categories:
    - ❖ Data Warehouse (also known ETL): There is three main steps as extractions, transformation, and loading as seen more details in below:
      - The extraction step connects to the sources for choosing and collecting the appropriate data for further analysis processing.
      - The transformation step aims to the extracted data (by some designed rules through the application) for converting in to a united format.
      - The loading step receives the extracted data first and then will send data to a target storage infrastructure.
    - ❖ Data Federation: This method builds a virtual database to query and aggregate data from different sources. In addition, the virtual database provides a container of information or metadata which is referred to the actual data and its location.
  - Cleansing: The data cleansing techniques provide a specific process to recognize inaccurate, incomplete, or unreasonable data. And then those data will be discarded or changed which improve the quality level of data. The cleansing techniques must have five main steps as shown as below:
    - ❖ Describe and recognize type of errors;
    - ❖ Exploring and identify error examples;
    - ❖ Solve the errors;
    - ❖ Register error reported and their types;
    - ❖ Correct the way of data entry procedures to increase any further errors in future.
  - Redundancy elimination: The data redundancy elimination techniques try to remove redundant datasets. The redundant data imposed more pressure for data transmission,

data storage, data processing, and so on. Therefore, many researchers proposed some techniques for redundancy detection, and data compression.

- In IoT, data quality problem can be referred in six main types as shown as below [180]:
  - Dropped readings: The ratio of successful delivery, between pervasive applications and their requested readings required, is reported with a minimum efficiency degree because of limited resources unstable communications. For instance, in [184] measured the dropping ratio.
  - Unreliable reading: There is some reason to be data unreliable such as impreciseness, calibration failure, fail dirty nodes and etc.
  - Multi-source data inconsistences: Handling IoT data is quite tough because IoT data is generated a number of different IoT devices (such as RFID, sensors and etc.) and with abundant data structured, unstructured, and semi-structured formats (such as text, image, numerical and etc.).
  - Data duplication: IoT devices installed everywhere to sense environments as much as they can. However, it sounds good but it makes an important challenge for IoT data which imposed by a number of redundant (duplicated) data. The duplicated data provides more costs for storage, processing, transmission, and etc.
  - Data leakage: It means that applications are retrieved and/or stored more data than their requirements. So, this problem imposed different challenges for IoT like user´s privacy.
  - Multi-source data time alignment: There are number of applications/services in IoT environments which are integrated multiple data sources. In fact, it appears some complexities and difficulties to extract appropriate data in terms of time-alignment (real time and/or historical data).
- In cloud computing, data quality and their related techniques (like data cleaning) acquire in cloud computing infrastructure [185].
- In smart city scenario, there are distinct devices and data formats providing different level of data quality. Data quality measurement usually is related to applications in one side [186]. On the other side, data quality of each data source can be assumed as on three depended factors [186]:
  - Data collection, from devices, is inaccurate;
  - A noisy environment and poor quality of data communication and processing can imposed problem for data quality
  - All observations and measurements are granularity in both spatial and temporal dimensions

In our view, the Data Quality phase aims to appraise the quality level of collected data. It is responsible for guaranteeing both, Quality Control (QC) and Quality Assurance (QA), in particular:

- Checking the quality level of data and discarding or repairing low quality data, according to the provided policies (QC).
- Monitoring the quality of data flows and, in case of continuous failures, proceeding according to the provided policies (QA).

5.1.3.3Objectives and challenges of an effective Data Quality phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data quality phase as part of a data management architecture. However, some of the quality challenges are out of scope of this research thesis, such as appraising the application quality, because our architecture has been designed for managing data and we assume applications are not under the control of the architecture. The main objectives and challenges are the following:

- Checking the quality level of data and discarding or repairing low quality data, according to the provided policies (QC). The particular quality methods required will depend on the city requirements.
- Monitoring the quality of data flows and, in case of continuous failures, proceeding according to the provided city management policies (QA).
- Additional difficulties arise when appraising data qualities when the data is getting large in volume [178].In addition, poor quality of data could impose irreparable damage to economy and society [14] because it can be acquired directly or indirectly incorrect analytic results, wrong decision making, and so on.
- Inaccurate data collection can be imposed some irreparable damage for any scenario such as producing invalid data analysis and results [155]. Plus, data collection must be matched with the objective of each particular data analysis.
- Covering some additional data problems, such as “dropped readings”, “unreliable reading”, “multi-source data inconsistences”, “data duplication”, “data leakage”, “multi-source data time alignment”, and so on.
- These objectives contribute in simplifying the data challenges of :
    - Data Veracity, because applying successfully the data quality techniques may help improving the veracity of data.
    - Data Value, because an increase in data quality indeed impacts in its value.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. However, it is out of the scope of this research thesis proposing new data appraising methods. We just show how easy and efficient is applying existing algorithms to this architecture. This will be described and discussed in subsection 5.2.

5.1.4 The Data Description phase

Figure 5.7 depicted that Data Description phase is the ultimate phase in the Data Acquisition block which we will describe more in the following subsections.

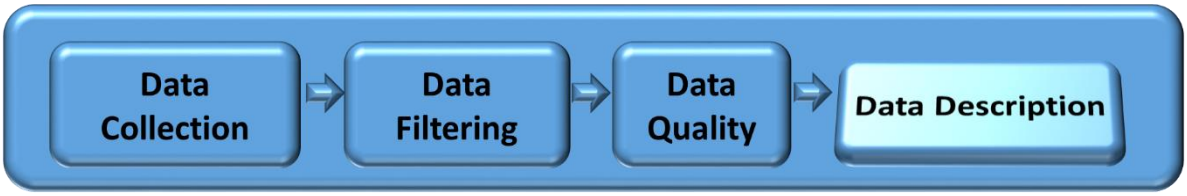


Figure 5.7 The Data Description phase in the Data Acquisition Block

5.1.4.1 Definition

The Data Description phase is mainly related to the concepts of Metadata definition and organization. The Metadata defines a structured representation of the data and provides contextual and higher value information. This representation is related to some more description of the data (including the produced data) which it can be useful for future usage of this data [187].

5.1.4.2 State of the art

Data description normally can be depicted with the concept of Metadata. In fact, Metadata is used for description of digital from several years ago. Besides that there is number of standards for Metadata description (for instance Dublin Core) which mainly refer to specific data sources (like web-based data). In addition, Metadata has two different types as shown as below [188]:

- Long-term digital archives: It means that Long-term digital archives normally keep all metadata received to them with corresponding digital objects. And then this long term-digital archives are tried to build a subset of obtained metadata to organize archive assets.
- Preservation metadata: This kind of metadata defines processes and preservation actions applied to digital objects in the archives.

There is some related work about data description in different scenario and environments:

- In general, the author said that Metadata is able to presented in data models (for instance, the BBC´s SMEF TM) or description schemes (e.g. MPEG-7 and Dublin Core) or professional domain (for instance combination of SMPTE) or combination of media and Metadata (e.g. MXF or BWF) and so on [187].
- In Sensors Network, the author [189] discussed that traditional data management focus on metadata and data as two separate entities. So, there is not any support for joint real-time processing of metadata and sensor data. However, currently there is some view about “distributed metadata joint”. So, Global Sensor Networks (GSN) and Sensor Metadata Repository (SMR) proposed to handle Metadata in federated sensors networks.

- In Smart City, the author presented the Metadata in the Open Data platform through a standard OAI-PMH protocol. This solution get helps for data fusion between Smart Cities which are able to grab data by other data platforms or search engines [190].
- In Fog-to-Cloud computing, the author mentioned that it is possible to create and store metadata in the edge of network through Fog-to-Cloud scenario. So, the base of this idea (IPFS) has been built on top of the BitTorrent protocol and KadmeliaDHT which make facility to apply Metadata management in the edge of networks[191].

5.1.4.4Objectives and challenges of an effective Data Description phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data description phase as part of a data management architecture. The main objectives and challenges are the following:

- Defining the appropriate metadata required according to the business model, such as timing information (creation, collection, modification, among others), positioning (city, country, GPS coordinates), authoring, privacy, and so on.
- Tagging all collected data with the corresponding description for easing future retrievals and analyses, in order to facilitate raw data interpretation.
- In case that standardization or normalization is required, some policies should be applied in order to guarantee the proper described of the produced data. Similarly, if reproducibility is required, an appropriate data description must be performed.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. However, it is out of the scope of this research thesis proposing new data description methods. We just assume that the particular city business model defines the metadata requirements. This will be described and discussed in the next subsection.

5.2 The Data Acquisition Block in the F2CSmart City

As shown in the previous chapter (Figure 4.5), the F2C data management architecture is able to efficiently handle the Data Acquisition block. So, the Data Acquisition block can be applied in Fog-Layer-1, Fog-Layer-2, and Cloud layer. The point is that each layer has the different computational capacity for applying the Data Acquisition block tasks and their related phases. In this section: i) first we show that the phases of the Data Acquisition block can be managed at different layers in the F2C data architecture; and ii) we present some details of the hierarchal distributed data optimization and filtering techniques which can be easily implemented as part of the Data Acquisition block and their related phases The data acquisition block in the F2C model can be seen in Figure 5.8.

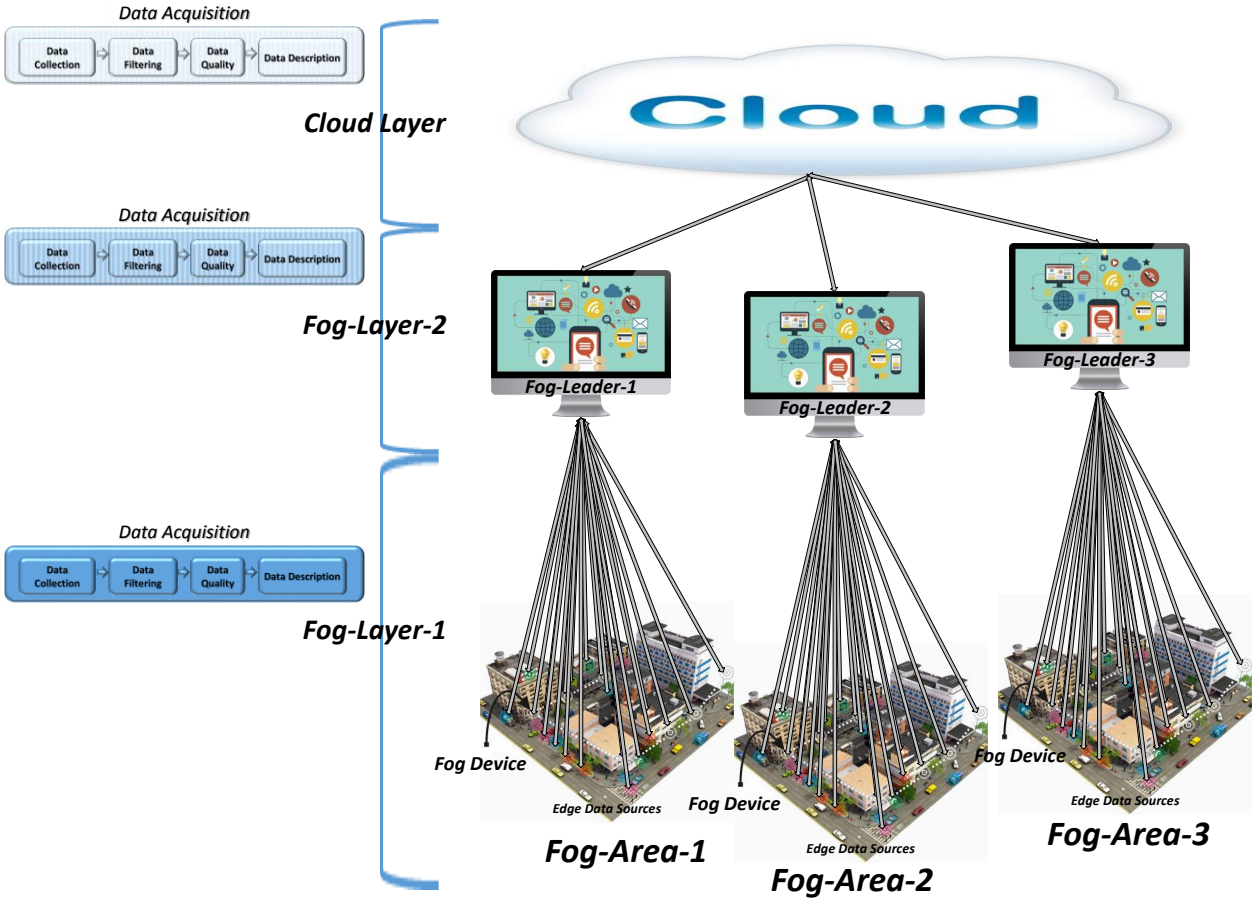


Figure 5.8 Description Scenario of the Data Acquisition Block

The Fog-Layer-1 is the main responsible for handling most of the related tasks in the Data Acquisition block in the F2C architecture. Similarly, the architecture is organized to manage the rest of the related task through Fog-Layer-2 and cloud layer. In addition, we assume that cloud is just responsible to handle the minimum amount of tasks related to data acquisition, instead of the traditional model[22]. So, we will explain some more details of the phases in the Data Acquisition block with tailoring to F2C in the below sentences.

- Fog-Layer-1:
  - The set of Edge-data-sources (as part of Fog-Layer-1) is mainly responsible for the Data Collection phase. It means that all possible data will be sensed by the IoT devices in the city. And then, this all data will be transferred to their corresponding Fog-Device for further preprocessing techniques.
  - The set of Fog-Devices (as another part of Fog-Layer-1) aims to provide some facility to handle the first level of data optimization models (such as data aggregation techniques) for refining and cleaning dark data among all collected data. In addition, data compression can apply in this layer to compress the refined data. And then, the compressed data will be sent to the upper layer through the network communication.

This layer is also able to check some level of data quality for collected data. In case, if the data does not meet the requirements to move to the next layer, the Fog-Device can register this history and look for solving this problem. If the problem does not solve in this layer, the data will be discarded in this layer. For example, in some cases the sensors or their communication faced with some problem (including calibration problem or network faults and attacks). So, in this case, the Fog-Device can detect this problem with exploring invalid data and can send the alert to the system manager for solving this abnormal problem. Plus, there is a possibility for Fog-Device to block the sensor node to prevent increasing any future faults.

This layer also is able to get help to the Data Description phase for tagging some metadata information over the collected data (such as ownership, data production date, and etc.).

- Fog-Layer-2:
  - The Fog-Leader is responsible for applying some high-level degree of tasks for most of data acquisition phases' inquiries (including data filtering techniques, data quality, and data description). However, there is not any data collection tasks in this layer.

    The second level of the data filtering tasks (such as data aggregation, compression and so on) can be applied in this layer. Note that in this layer, the Fog-Leader receives collected data from all Fog-Devices belonging to his Fog-Area, so the Fog-Leader can easily implement some additional data aggregation techniques, and combine redundant information collected from the nodes in his area.

    Similarly, data quality is active in this layer. We have the higher level of checking quality in this layer for detecting and discarding the poor data. Additionally, the poor data will be discarded and reported in the system.

    In this layer, there is a possibility to add some more additional information to the related Metadata (such as data source path and etc.).
- Cloud Layer:
  - Cloud technologies provide a splendid facility to complete all reminded tasks for all phases of the Data Acquisition block.

The global behavior is as follows. Data are mainly collected al Fog-Layer-1 through the city sensors network. Unlike in other architectures where data are moved immediately to a centralized cloud (or similar) platform, data can be kept in this low layer. If a real-time or critical data access is required, data will be immediately provided with such layer, thus reducing considerably the latency and therefore providing optimal performance. Then the data will be filtered and appraised to meet the quality standards required. This process is performed at the same Fog-Layer-1 and can be as complex and sophisticated as desired because data are still available from there at any time. After some time (configurable according to the city business model) the data is transferred to the following upper layer. In the experimental section we show that data can be aggregated and compressed up more than 90% of their original size, so the amount of data that will be transferred

upwards is drastically reduced. In addition, note that such data transfer can be delayed as required, so an additional optimization that can be made is to send the data when the network traffic is low. This process is repeated at each fog layer until, finally, all data (aggregated and compressed, meeting high quality standards) reaches the cloud layer. In this layer all data is preserved and ready for future retrieval of historical data.

The advantages of this data acquisition architecture are numerous, and can be summarized as follows:

- Data can be effectively collected distributed throughout the whole city sensors network.
- Data is ready to be used immediately and benefit from the advantages of locality, as opposed to a centralized platform where data must be read first to the cloud, and then, accessed from the cloud (two times a long distance transfer).
- Data is effectively organized for future retrievals.
- The amount of data to be transferred to higher layers is less than 10% of the total data collected thanks to the filtering phase, which efficiently aggregates and compresses the data.
- Data transfer all higher layers can be done when the network load is low, therefore optimizing the global network traffic.
- The data filtering and data quality processes are not limited in time, as long as data are already accessible from the lower layers and, therefore, any sophisticated or complex process can be performed.

As can be seen, we can conclude that all objectives and challenges listed in each phase of this block have been successfully addressed in this data management architecture.

5.3Experimental results: Estimating Data Acquisition in Barcelona

In this section, first, we propose our F2C architecture for Barcelona city with respect to the main F2C architecture in the previous chapter (in Figure 4.10). Then, we describe the sensors data through Sentilo platform in Barcelona (including the data type and their related category). And, we appraise the current sensors data collection (including our methodology and results) in Barcelona city. And then we estimated the future data collection (consisting of our methodology and results) in Barcelona Smart City (including sensors and other types of data). Next, we described our data filtering measurement through data collection (including our methodology and results) in Barcelona Smart City. Then, we explained our data compressing measurement through data collection (including our methodology and results) in Barcelona Smart City. Indeed, we discussed our results in this section.

5.3.1F2C architecture for Barcelona

The F2C architecture presented in Figure 4.10 was a three layer basic model (Layer-1, Layer-2, and cloud) used to easily show the data management concepts in a simple architecture. However, the hierarchical architecture allows a flexible number of layers according to the city structure or the business model requirements. In this section, we aim to draw the F2C architecture for Barcelona Smart City.

First, we focus about the urban structure of the city. As shown in Table 5.3Barcelona has ten main districts and seventy-three sections [192].In fact each main district covers by some main sections. For example, Ciutat Vella includes with four sections. So, as we said before, we make a match between our scenario (Figure 5.10) and the number of districts and sections (as shown in Table 5.7) in Barcelona city. So, each Fog-Area (in Fog-layer-1) represents one section of Barcelona city in one side. A number of sections (variable, depending on the district) are organized to become a district, so the corresponding Fog-Leaders (of each section) are grouped (in Fog-Layer-2) and managed by a Fog-Leader at to create the district (in Fog-Layer-3). Finally, the set of districts (all Fog-Leaders at layers 3) are grouped in the cloud-layer and therefore covering the whole city.

Table 5.3 Numbers of districts and sections in Barcelona city

Number	Distericts	Sections
1	Ciutat Vella	el Raval
2		el Barri Gòtic
3		la Barceloneta
4		Sant Pere, Santa Caterina i la Ribera
5	Eixample	el Fort Pienc
6		la Sagrada Família
7		la Dreta de l'Eixample
8		l'Antiga Esquerra de l'Eixample
9		la Nova Esquerra de l'Eixample
10		Sant Antoni
11	Sants-Montjuïc	el Poble Sec (1)
12		la Marina del Prat Vermell (2)
13		la Marina de Port
14		a Font de la Guatlla
15		Hostafrancs
16		la Bordeta
17		Sants - Badal
18		Sants
19	Les Corts	les Corts
20		la Maternitat i Sant Ramon
21		Pedralbes
22	Sarrià-Sant Gervasi	Vallvidrera, el Tibidabo i les Planes
23		Sarrià
24		les Tres Torres
25		Sant Gervasi - la Bonanova
26		Sant Gervasi - Galvany
27		el Putxet i el Farró
28	**Gràcia**	Vallcarca i els Penitents
29		el Coll
30		la Salut
31		la Vila de Gràcia
32		el Camp d'en Grassot i Gràcia Nova
33	Horta-Guinardó	el Baix Guinardó
34		Can Baró
35		el Guinardó
36		la Font d'en Fargues
37		el Carmel
38		la Teixonera
39		Sant Genís dels Agudells
40		Montbau
41		la Vall d'Hebron
42		la Clota
43		Horta
44	Nou Barris	Vilapicina i la Torre Llobeta
45		Porta
46		el Turó de la Peira
47		Can Peguera
48		la Guineueta
49		Canyelles
50		les Roquetes
51		Verdun
52		la Prosperitat
53		la Trinitat Nova
54		Torre Baró
55		Ciutat Meridiana
56		Vallbona
57	Sant Andreu	la Trinitat Vella
58		Baró de Viver
59		el Bon Pastor
60		Sant Andreu
61		la Sagrera
62		el Congrés i els Indians
63		Navas
64	Sant Martí	el Camp de l'Arpa del Clot
65		el Clot
66		el Parc i la Llacuna del Poblenou
67		la Vila Olímpica del Poblenou
68		el Poblenou
69		Diagonal Mar i el Front Marítim del Poblenou
70		el Besòs i el Maresme
71		Provençals del Poblenou
72		Sant Martí de Provençals
73		la Verneda i la Pau

With respect to the urban structure of Barcelona city, we realized that our proposed architecture must have four layers regarding the number of districts and sections of Barcelona as shown in Table 5.7. Those four layers namely is called Fog-Layer-1, Fog-Layer-2, Fog-Layer-3, and cloud layer as shown in Figure 5.9. Plus, Fog-layer-2 assumes to be like numbers of the sections in Barcelona. Similarly, Fog-Layer-3 considers like numbers of the districts in Barcelona.

Similarly, Fog-Leader-A (in Fog-Layer-2) is able to communicate with Fog devices in Barcelona Smart City. Additionally, Fog-Leader-1 (in Fog-Layer-3) can be contacted with the different numbers of Fog-Leader-A (in Fog-Layer-2).

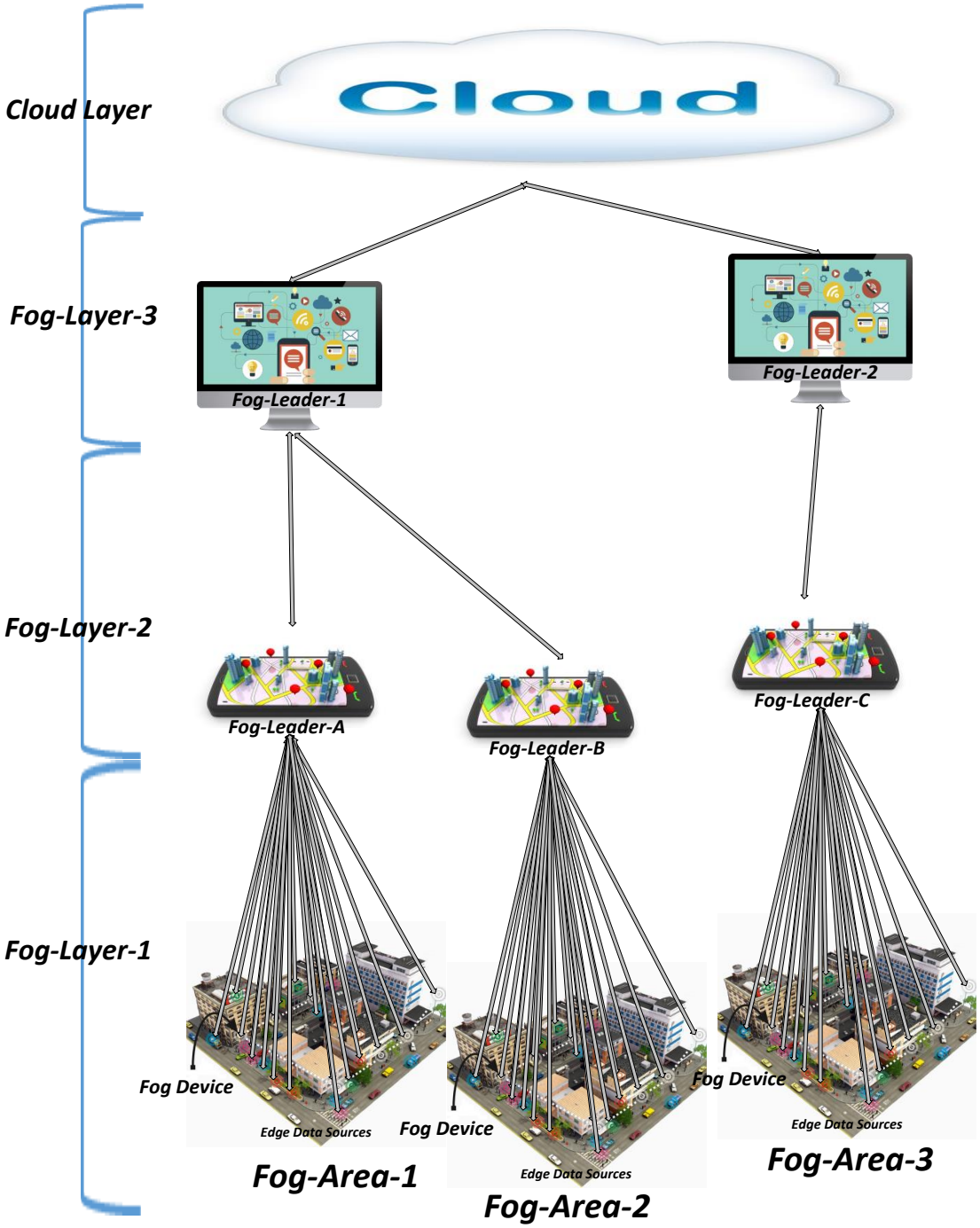


Figure 5.9 Description Scenario of Barcelona Smart City

5.3.2 Description of the sensors' deployment in Barcelona

As explained in the previous chapter, Sentilo platform provides a piece of sensors data collection in Barcelona. Nowadays, Sentilo platforms generates five categories of information in the city which is namely Energy monitoring, Noise monitoring, Urban Lab monitoring and Garbage Collection monitoring and Parking Spots monitoring as shown in Figure 5.10. Recently, Sentilo has plan to extend their sensors network with two objectives. First, the Sentilo likes to have full coverage in all districts and zones of Barcelona city with their current types of sensors and information. And then, the Sentilo willing to catch more categories of information to sense the Barcelona city more than before like water meter, camera data and etc. So, in this section, we aim to calculate the current produced sensors data by Sentilo platform and analyze the amount, type, and frequency of the generated data by each type of sensor. And then, we will measure a number of sensors data for the future Sentilo platform and Barcelona Smart City and refine some more information to utilize for the data preservation block.

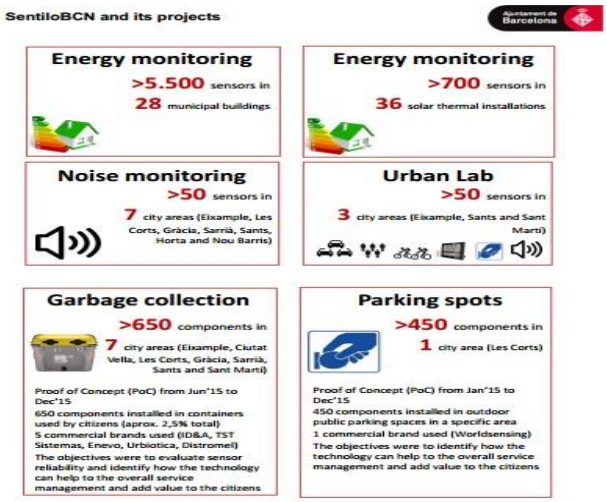


Figure 5.10 Category of produced information through Sentilo platform

In Sentilo platform, each category of information included with different type of information to cover objectives of their related category as shown in below:

- Energy monitoring category:
  - Electricity meter
  - External ambient conditions
  - Gas meter
  - Internal ambient conditions
  - Network analyzer
  - Solar thermal installation
  - Temperature

- Noise monitoring category:
  - First type of noise:
  - Second type of noise
  - Third type of noise
- Urban Lab monitoring management category:
  - Air quality
  - Bicycle flow
  - People flow
  - Traffic
  - Weather
- Garbage Collection monitoring management
  - Glass container
  - Organic container
  - Paper container
  - Plastic container
  - Refuse container
- Parking Spots monitoring management:
  - Parking

5.3.3Estimating Current data collection in Barcelona

In 2016, Sentilo installed 1,800 sensors in the city of Barcelona. Those installed sensors are able to collect information in five categories of information as shown in aforementioned sentences (including Energy, Noise, Urban Lab, garbage container and parking slots). Now, we aim to calculate the number of sensors data collection in Barcelona city (including current sensors data by Sentilo platform, future sensors data by Sentilo platform, and future data collection).

In this section, first, we introduce about our methodologies to calculate the current sensors data collection by Sentilo platform in Barcelona. And then, we will calculate and show the numbers of the sensor data collection for each category of information through Sentilo platform in Barcelona city.

5.3.3.1 Methodology

1-Realizingsome sensors data information:

For each type of information, we get initial information about the current Sentilo platform as shown in in Table 5.4 (including the number of sensors devices, the different type of information, and frequency of sending and updating information).

Energy monitoring category covers the different type of information (including electricity meter, external ambient conditions, gas meter, internal ambient conditions, network analyzer, and solar thermal installation temperature). Plus this category of information installed with 541 physical sensors in the Barcelona city (including 28 sensors for "electricity meter", 7 sensors for "external ambient conditions", one sensor for "gas meter", 41 sensors for "internal ambient conditions", 421 sensors for "network analyzer", 36 sensors for "solar thermal installation", and 7 sensors for "temperature"). Additionally, the frequency of sending and updating information would be different for each type of information in this category. In most of the case, the sensors data send and update their related information in every 15 minutes (for average data) or one minute (for instantaneous data).

Noise monitoring category has three different type of information. This category covers by 53 sensors in the city (including 3 for the first type of noise, 40 sensors for the second type of noise, and 10 sensors for the third type of noise information). In addition, the frequency of update and sending information has designed in every 15 minutes (for the first type of noise information) and every one minute (for the second and third type of information).

Urban Lab monitoring category proposed with the different type of information (including air quality, bicycle flow, people flow, traffic, and meteo) on one side. On the other side, twenty-one sensors are installed in the city (encompassing four sensors for air quality, two sensors for bicycle flow, four sensors for traffic, and seven sensors for meteo information). Further, there is specific rules for sending and updating information in this category. The rules are every 30 minutes (for meteo information), every 15 minutes (for air quality information), every 10 minutes (for people and bicycle flow information), and every one minute (for traffic information).

Garbage collection category comes with different type of information which are glass container, organic container, paper container, plastic container, and refuse container. And, all those information sensed by 667 sensors in the Barcelona city (including 57 sensors for the glass container, 71 sensors for the organic container, 57 sensors for the paper container, 205 sensors for the plastic container, and 277 sensors for the refuse container). In addition, this policy of update and sending information are same for all types of information which occur every 20 minutes or every 60 minutes.

Indeed, Parking Spot category includes with parking information. There are 513 sensors in the city to sense the related information about parking spaces. In addition, the information will be sent and update every seven hours. Plus, in case if there is any change in the status of information, the information will be sent and updated immediately.

Table 5.4All categories of Sentilo sensors data in the current Smart City of Barcelona

Type	Number of devices	Frequency of sending and updating information
Electricity meter	28	Every 1 minute (instantaneous data) and every 15 minutes (average data)
External ambient conditions	7	Every 1 minute (instantaneous data) and every 15 minutes (average data)
Gas meter	1	Every 1 minute (instantaneous data) and every 15 minutes (average data)
Internal ambient conditions	41	Every 1 minute (instantaneous data) and every 15 minutes (average data)
Network analyzer	421	Every 1 minute (instantaneous data) and every 15 minutes (average data)
Solar thermal installation	36	Every 15 minutes
Temperature	7	Every 15 minutes
Total	541	

(a) Energy monitoring

Type	Number of devices	Frequency of sending and updating information
Noise	3	Every 15 minutes
	40	Every 1 minutes
	10	Every 1 minutes
Total	53	

(b) Noise monitoring

Type	Number of devices	Frequency of sending and updating information
Air quality	4	Every 15 minutes
Bicycle flow	2	Every 10 minutes
People flow	4	Every 10 minutes
Traffic	4	Every 1 minute
Meteo	7	Every 30 minute
Total	21	

(c) Urban Lab monitoring

Type	Number of devices	Frequency of sending and updating information
Container glass	57	Every 20 minutes or Every 60 minutes
Container organic	71	Every 20 minutes or Every 60 minutes
Container paper	57	Every 20 minutes or Every 60 minutes
Container plastic	205	Every 20 minutes or Every 60 minutes
Container refuse	277	Every 20 minutes or Every 60 minutes
Total	667	

(d) Garbage Collection monitoring

Type	Number of devices	Frequency of sending and updating information
Parking	513	Every change of status and (always) every 7 hours
Total	513	

(e) Parking Spots monitoring

2- Calculating the produced data for each sensor per each transaction:

For each type of information, we calculate how many data will be produced by each sensor per transaction. In this case, we assume that each type of data (in terms of the float or integer number) transfers 4 bytes or 2 bytes in fact. So, for instance as shown in Figure 5.11, the sensors data (for electricity meter) transfer 22 byte for each transaction because regarding Sentilo platform, each electricity meter sensors data attached with five float data (such as electricity meter, time, date, and etc.) and a single integer data (location code).

(Float) numeric data type	(Float) numeric data type	(Float) numeric data type	(Float) numeric data type	(Float) numeric data type	(Integer) numeric data type

Figure 5.11Sensors Data Transfer Packet (Electricity Meter)

3- Calculating the produced data for each sensor per day:

Now, we aim to calculate the number of the produced data for each sensor per day as shown in Table 5.9. So, now we have two main parameters (number of the produced data per each transaction by each sensor and the frequency of sending and update information for each type of information). So, first, we must calculate data how frequent updating and sending information will happen per hour. And then, we can calculate easily the total produced data by the below formula:

Total Produced Data by each sensor per day = Produced data per transaction * Frequency of sending and update information per hour

4- Calculating the produced data for all sensors per day:

Now, we aim to calculate how many data will be produced by all sensors per day as shown in Table 5.9.Similarly, we have two main parameters (total number of sensors devices and Total

Produced Data by each sensor per day). So, we can calculate all the produced data per day in below formula.

All Produced Data = Total numbers of sensors devices* Total Produced Data by each sensor per day

5- Calculating the produced data for each category of sensors data per transaction:

Similarly, as shown in Table 5.8, we aim to calculate how many data will be generated by all sensors for each category of sensors data in every single day. So, if we just accumulate all produced data for each category of information per transaction, we can reach to this number.

6- Calculating the produced data for all categories of sensors data in every single day:

Now, we would calculate how many data will be generated for all categories of information (all five categories of Sentilo data) in every day. So, we just need to accumulate all calculated number for each category of sensors data in every single day (as discussed in the previous description).

5.3.3.2Results

There is five category of information which we will show how much data will be produced for each of them as shown in below:

- Energy monitoring category:

  The number of sensors (for each category)is as shown in below:

  - Electricity meter: 28 sensors
  - External ambient conditions: 7sensors
  - Gas meter: 1sensor
  - Internal ambient conditions: 41sensor
  - Network analyzer: 421sensors
  - Solar thermal installation: 36sensors
  - Temperature: 22 sensors

  As shown in Table 5.9, the data production (by each sensor at each transaction) is shown in below:

  - Electricity meter: 22 byte per transaction
  - External ambient conditions: 22 byte per day
  - Gas meter: 22 byte per day
  - Internal ambient conditions: 22 byte per day
  - Network analyzer: 242 byte per day
  - Solar thermal installation: 22 byte per day

- Temperature: 22 byte per day

So, the total amount of the production data is 374 byte per transaction.

As shown in Table 5.9, the data production (by each sensor per day) is shown in below:

- Electricity meter: 2,112 byte per day
- External ambient conditions: 2,112 byte per day
- Gas meter: 2,112 byte per day
- Internal ambient conditions: 2,112 byte per day
- Network analyzer: 23,232 byte per day
- Solar thermal installation: 2,112 byte per day
- Temperature: 2,112 byte per day

Therefore, the total amount of data is 35,904 byte which it must be transferred by each sensor at each transaction.

As shown in Figure 5.12, the total data production per day is listed in the below for each type of information in this category.

- Electricity meter: 59,136 byte per day
- External ambient conditions: 14,784 byte per day
- Gas meter: 2,112 byte per day
- Internal ambient conditions: 86,592 byte per day
- Network analyzer: 9,780,672 byte per day
- Solar thermal installation: 76,032 byte per day
- Temperature: 14,784 byte per day

As we shown in Table 5.5, the total amount of data production is 11MB per day which it must be transferred to the upper layer for further usage.

Table 5.5 Produced sensors data through Sentilo in current Barcelona city

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Electricity meter	28	22	2,112	59,136
External ambient conditions	7	22	2,112	14,784
Gas meter	1	22	2,112	2,112
Internal ambient conditions	41	22	2,112	86,592
Network analyzer	421	242	23,232	9,780,672
Solar thermal installation	36	22	2,112	76,032
Temperature	7	22	2,112	14,784
Total	541	374	35,904	10,034,112

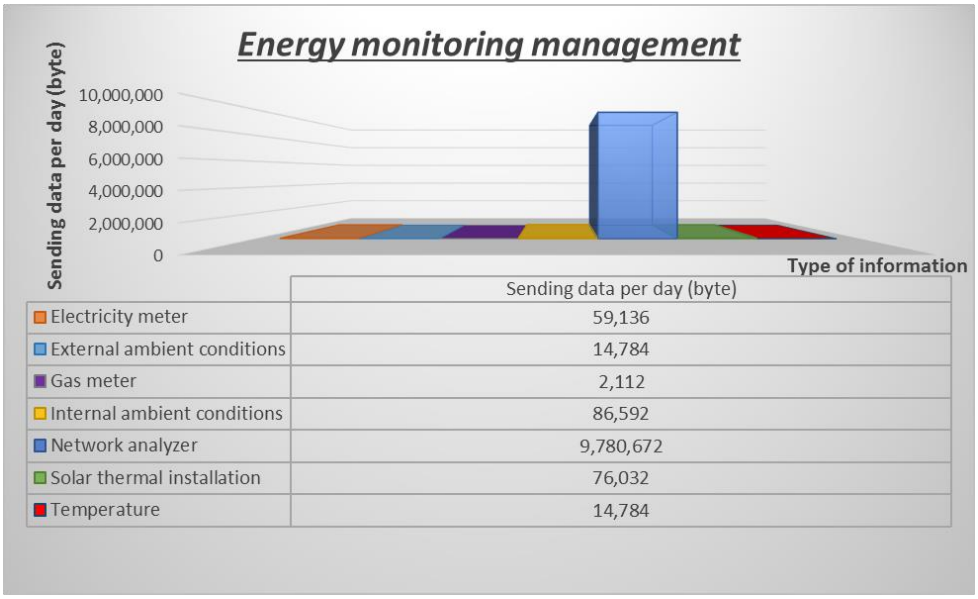


Figure 5.12 Sending daily data in Energy monitoring category

- Noise monitoring category:

  The number of sensors (for each category) is as shown in below:

  - First type of noise: 3 sensors
  - Second type of noise: 40 sensors
  - Third type of noise: 10 sensors

As shown in Table 5.6, the data production (by each sensor per day) is shown in below:

- First type of noise: 22 byte
- Second type of noise: 22 byte
- Third type of noise: 22 byte

So, the total amount of the production data is 66 byte per transaction.

As shown in Table 5.6, the data production (by each sensor at each transaction) is shown in below:

- First type of noise: 2,112 byte
- Second type of noise: 31,680 byte
- Third type of noise: 31,680 byte

Therefore, the total amount of data is 65,472 byte which it must be transferred by each sensor at each transaction.

As shown in Figure 5.13, the total data production per day is listed in the below for each type of information in this category.

- First type of noise: 6,336 byte
- Second type of noise: 1,267,200 byte
- Third type of noise: 316,800 byte

As we shown in Table 5.6, the total amount of data production is almost 1 MB byte per day which it must be transferred to the upper layer for further usage.

Table 5.6 Produced sensors data through Sentilo in current Barcelona city

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Noise	3	22	2112	6336
	40	22	31,680	1,267,200
	10	22	31680	316800
Total	53	66	65,472	1,590,336

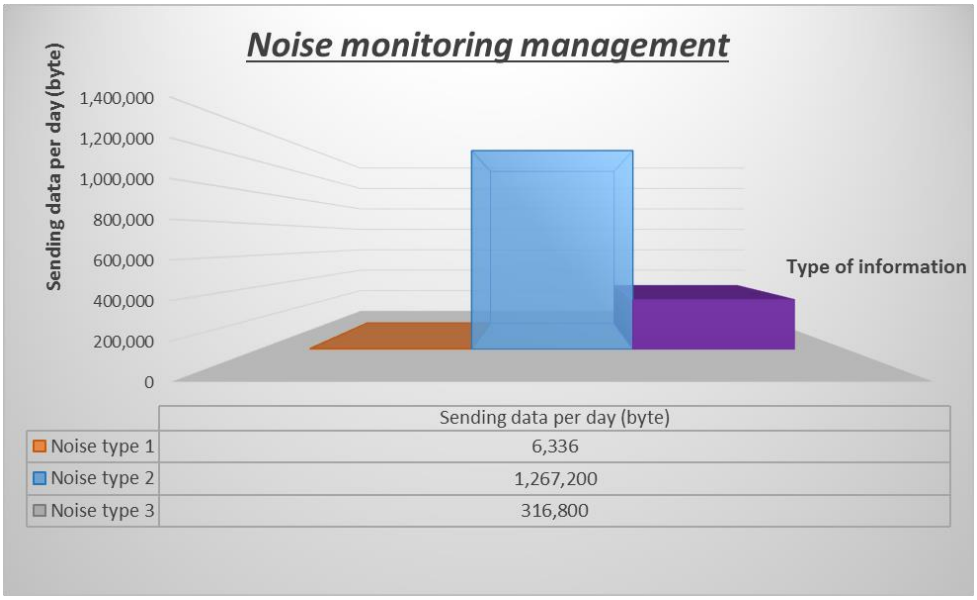


Figure 5.13 Sending daily data in Noise monitoring category

- Urban Lab monitoring management category:

  The number of sensors (for each category) is as shown in below:

  - Air quality: 4 sensors
  - Bicycle flow: 2 sensors
  - People flow: 4 sensors
  - Traffic: 4 sensors
  - Weather: 7 sensors

  As shown in Table 5.7, the data production (by each sensor at each transaction) is shown in below:

  - Air quality: 144 byte
  - Bicycle flow: 22 byte
  - People flow: 22 byte
  - Traffic: 44 byte
  - Weather: 120 byte

  So, the total amount of the production data is 352 byte per transaction.

  As shown in Table 5.7, the data production (by each sensor per day) is shown in below:

  - Air quality: 13,824 byte
  - Bicycle flow: 3,168 byte
  - People flow: 3,168 byte
  - Traffic: 63,360 byte

- Weather: 34,560 byte

Therefore, the total amount of data is 118,080 byte which it must be transferred by each sensor at each transaction.

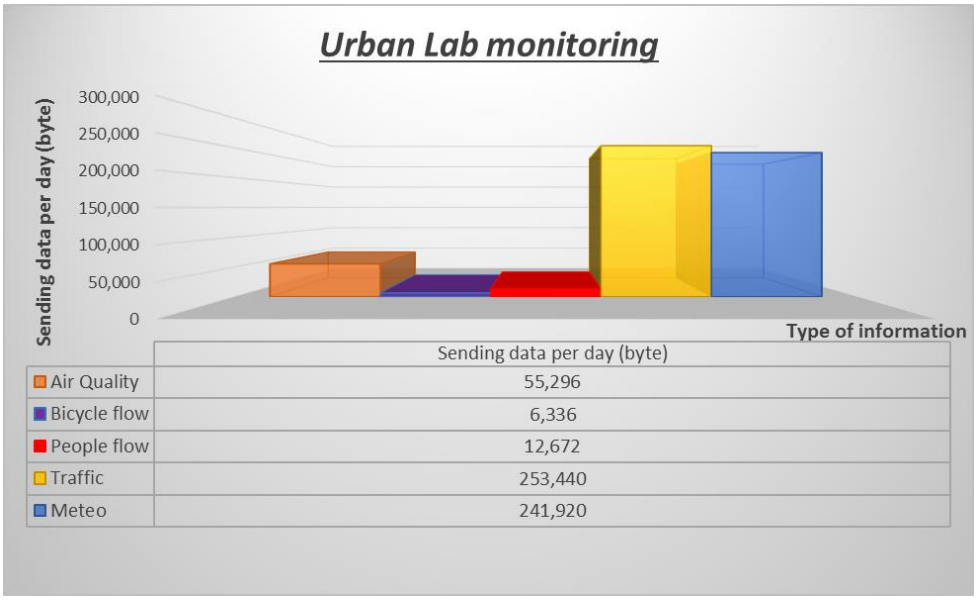
As shown in Figure 5.14, the total data production is listed in the below for each type of information in this category.

- Air quality: 55,296 byte
- Bicycle flow: 6,336 byte
- People flow: 12,672 byte
- Traffic: 253,440 byte
- Weather: 241,920 byte

As we shown in Table 5.11, the total amount of data production is 569,664 byte per day which it must be transferred to the upper layer for further usage.

Table 5.7 Produced sensors data through Sentilo in current Barcelona city

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Air quality	4	144	13,824	55296
Bicycle flow	2	22	3,168	6,336
People flow	4	22	3,168	12,672
Traffic	4	44	63,360	253,440
Weather	7	120	34,560	241,920
Total	21	352	118,080	569664



	Sending data per day (byte)
Air Quality	55,296
Bicycle flow	6,336
People flow	12,672
Traffic	253,440
Meteo	241,920

Figure 5.14 Sending daily data in Urban Lab monitoring category

- Garbage Collection monitoring management

  The number of sensors (for each category) is as shown in below:

  - Glass container: 57 sensors
  - Organic container: 71 sensors
  - Paper container: 57 sensors
  - Plastic container: 205 sensors
  - Refuse container: 277 sensors

  As shown in Table 5.8, the data production (by each sensor at each transaction) is shown in below:

  - Glass container: 50 byte
  - Organic container: 50 byte
  - Paper container: 50 byte
  - Plastic container: 50 byte
  - Refuse container: 50 byte

  So, the total amount of the production data is 250 byte per transaction.

  As shown in Table 5.8, the data production (by each sensor at per day) is shown in below:

  - Glass container: 1,800 byte
  - Organic container: 1,800 byte
  - Paper container: 1,800 byte
  - Plastic container: 1,800 byte

- Refuse container: 1,800 byte

Therefore, the total amount of data is 9,000 byte which it must be transferred by each sensor at each transaction.

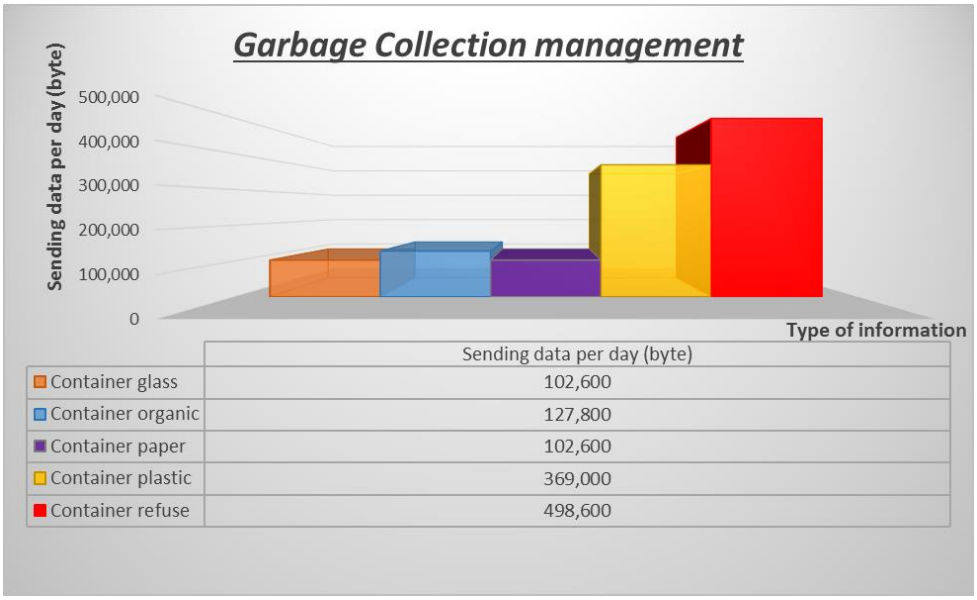
As shown in Figure 5.15, the total data production is listed in the below for each type of information in this category.

- Glass container: 102,600 byte
- Organic container: 127,800 byte
- Paper container: 102,600 byte
- Plastic container: 369,000 byte
- Refuse container: 498,600 byte

As we shown in Table 5.8, the total amount of data production is 1,200,600 byte per day which it must be transferred to the upper layer for further usage.

Table 5.8 Produced sensors data through Sentilo in current Barcelona city

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Container glass	57	50	1,800	102,600
Container organic	71	50	1,800	127,800
Container paper	57	50	1,800	102,600
Container plastic	205	50	1,800	369,000
Container refuse	277	50	1,800	498,600
Total	667	250	9,000	1200600



	Sending data per day (byte)
Container glass	102,600
Container organic	127,800
Container paper	102,600
Container plastic	369,000
Container refuse	498,600

Figure 5.15 Sending daily data in Garbage Collection monitoring category

- Parking Spots monitoring management:

  The number of sensors (for each category) is as shown in below:

  - Parking: 513 sensors

  As shown in Table 5.9, the data production (by each sensor at each transaction) is shown in below:

  - Parking: 40 byte

  So, the total amount of the production data is 40 byte per transaction.

  As shown in Table 5.9, the data production (by each sensor per day) is shown in below:

  - Parking: 4,000 byte

  Therefore, the total amount of data is 4,000 byte which it must be transferred by each sensor at each transaction.

  As shown in Figure 5.16, the total data production is listed in the below for each type of information in this category.

  - Parking: 2,052,000 byte

  As we shown in Table 5.9, the total amount of data production is 2,052,000 byte per day which it must be transferred to the upper layer for further usage.

Table 5.9 Produced sensors data through Sentilo in current Barcelona city

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Parking	513	40	4,000	2,052,000
Total	513	40	4,000	2,052,000

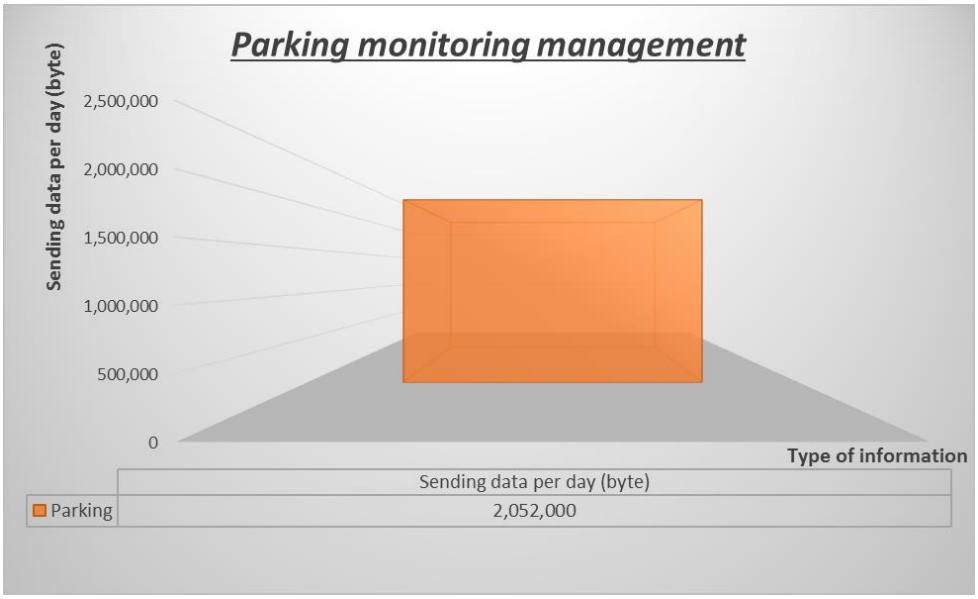


Figure 5.16 Sending daily data in Parking Spots monitoring category

We conclude that currently there are several types of sensors in the Barcelona which is able to sense and produce five categories of information with their related data type. Then, we have estimated the amount sensors and data to deploy a ubiquitous Smart City in Barcelona, as shown in Table 5.9, 5.10, 5.11, 5.12, and 5.13 (a, b, c, d, and e). Consequently, we have measured the total number of data to be about 8 GB (8,583,503,168) per day from all sensors in the city.

5.3.4Estimating Future data collection in Barcelona

As we described so far, currently the total number of data to be about 8 GB (8,583,503,168) per day from all sensors in the city. Moreover, we must consider 320,925,019 sensors for a whole coverage in the city. It is worth noticing that this volume only measures data obtained from sensors, and does not consider other data obtained from other eventual sources. So, in this section, we aim to extend the data type in the future of Barcelona city, including sensors data and other data types. And then we argue that the numbers of the produced data with both sensors data and other data types in the city of Barcelona.

5.3.4.1Methodology

Here first we present our methodology to estimate the future sensors data through Sentilo platform as shown in below:

1- Estimating the number of sensors devices:

According to the statistical department in the Barcelona's City Hall, there were 70,717 buildings, 40,000 containers, and 80,000 parking slots in 2014 [147, 193]. In addition, we have estimated that there are around 40,000 street corners in Barcelona. Thus, in this section, we present a projection of the number of sensors, and the corresponding expected generated data, assuming a complete sensors network deployment fully covering the Barcelona area. For instance with respect that information, we conclude that we must have 80,000 sensors for parking requirements in the Barcelona city as shown in Table 5.14.

2- Calculating all produced data per day:

As we said aforementioned, we have the number of sensors in our hand. On the other hand, we have also the number of sending data per each transaction and day for each category of information as shown in above sections. Therefore, we can calculate all produced data per day (with all sensors).

All Produced Data = Total numbers of sensors devices * Total Produced Data by each sensor per day

Next, we describe our methodology to add the new data type in future Smart City of Barcelona. And, we calculate the future data collection through those new data type in Barcelona city as shown in below:

1- Realizing new data types and category:

We checked some related work about other data type in another Smart City in the worldwide [194-200]. So, we realized that we can have two more main category of data (mobile application and camera) in Barcelona regarding our study in other Smart City and requirements of Barcelona city. Plus, energy monitoring management and vehicular mobility category can be added with new data types. Those are water meter and vehicular mobility data.

2- Estimating the number of sensors devices and generated data:

Here we used the official information about the equipment of Barcelona city to estimate the data number in the city of Barcelona as shown details in below:

Regarding energy monitoring management category, we aim to estimate the number of water consumption in Barcelona city. In [194, 195], the author mentioned that 5,242,880 byte data will be produced for 61,263 households in Surrey city of Canada. So, in [196] reported that the number of households in each district of Barcelona city. For example, there are 40,159 households is in Ciutat Vella. And then the total number of households are 654,979 households in Barcelona city.

Regarding urban monitoring management category, we aim to estimate the number of vehicular mobility data in Barcelona city. In [198, 199], the author reported that 700 cars can send 4.03 GB data per day in the city of Koln at Germany. So, it means that each car can produce 5,800,000 bytes (0.0058 GB) data per day. So, regarding the [200], there are 958,512 registered cars in Barcelona. And they mentioned that how many cars are available in each district. For example, there are 42,488 registered cars in Ciutat Vella. Therefore, we can estimate how much data will be produced in each district of Barcelona with comparing to information of Koln city.

Regarding one of one of the famous mobile application in Barcelona [201], they reported this mobile application will be produced 293.691 MB data in three years among 800 active users. So, it means that each user will be produced almost 351.55 byte data per day. Now we see that the Barcelona has 1,604,555 numbers of the population [197]. And then if we assume that we can have three type of mobile applications in terms of user usage (including high, medium, and the low-level range of user usage), we can assume below definition for this three type of mobile applications as shown as below:

- If our application has more than a million active users, we consider as the high-level range of user usage.
- If our application has a range of 500,000 to 1 million active users, we consider as medium level range of user usage.
- If our application has less than 500,000 active users, we consider as low level range of user usage.

Indeed, we estimate that the high-level range of application will be produced 435.84 MB data per day. Similarly, the medium level range of application will be produced 251.45 MB data per day. And, the low-level range of application will be generated 83.82 MB data in the single day.

Regarding camera data, in [202], the author said that each camera will be generated data around 60 to 120 MB per hour. So, we assume that each camera will be generated data around 90 MB in average. In [203], the author reported that there are 34,000 traffic lights in the city of Barcelona. So, we see that each traffic light has a single installed camera to capture the street in the city in one hand. On the other hand, we have 80,000 parking slots in Barcelona as we stated before. So each parking slots can be covered with a camera. Indeed, we count the number of cameras in Barcelona airport (Terminal 1). So, in [204] mentioned that there are 166 check-in desks, 15 baggage carousels, and 12,000 parking slots.

3- Frequency of sending and updating information:

If in case we need to have the frequency of sending and updating information (in particular for water meter data), we assume the frequency of sending and updating information in Sentilo platform. For example, the water meter follows the frequency of sending and updating information like other type of energy monitoring category which happens in every 15 minutes (for average data).

5.3.4.2Results

Here first we present our results for future sensors data through Sentilo platform as shown in below:

The number of sensor (for each category) is as shown in below:

- Electricity meter: 70,717 sensors
- External ambient conditions: 70,717 sensors
- Gas meter: 70,717 sensors
- Internal ambient conditions: 70,717 sensor
- Network analyzer: 70,717 sensors
- Solar thermal installation: 70,717 sensors
- Temperature: 70,717 sensors

As shown in Table 5.10, the data production (by each sensor at each transaction) is shown in below:

- Electricity meter: 22 byte per transaction
- External ambient conditions: 22 byte per day
- Gas meter: 22 byte per day
- Internal ambient conditions: 22 byte per day
- Network analyzer: 242 byte per day
- Solar thermal installation: 22 byte per day
- Temperature: 22 byte per day

So, the total amount of the production data is 374 byte per transaction.

As shown in Table 5.10, the data production (by each sensor per day) is shown in below:

- Electricity meter: 2,112 byte per day
- External ambient conditions: 2,112 byte per day
- Gas meter: 2,112 byte per day
- Internal ambient conditions: 2,112 byte per day
- Network analyzer: 23,232 byte per day
- Solar thermal installation: 2,112 byte per day
- Temperature: 2,112 byte per day

Therefore, the total amount of data is 35,904 byte which it must be transferred by each sensor at each transaction.

As shown in Figure 5.17, the total data production is listed in the below for each type of information in this category.

- Electricity meter: 149,354,304 byte per day
- External ambient conditions: 149,354,304 byte per day
- Gas meter: 149,354,304 byte per day
- Internal ambient conditions: 149,354,304 byte per day
- Network analyzer: 1,642,897,334 byte per day
- Solar thermal installation: 149,354,304 byte per day
- Temperature: 149,354,304 byte per day

As we shown in Table 5.10, the total amount of data production is 2,539,023,168 byte per day which it must be transferred to upper layer for further usage.

Table 5.10 Produced energy monitoring sensors data through Sentilo in future Barcelona city

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Electricity meter	70,717	22	2,112	149,354,304
External ambient conditions	70,717	22	2,112	149,354,304
Gas meter	70,717	22	2,112	149,354,304
Internal ambient conditions	70,717	22	2,112	149,354,304
Network analyzer	70,717	242	23,232	1,642,897,344
Solar thermal installation	70,717	22	2,112	149,354,304
Temperature	70,717	22	2,112	149,354,304
Total	495,019	374	35,904	2,539,023,168

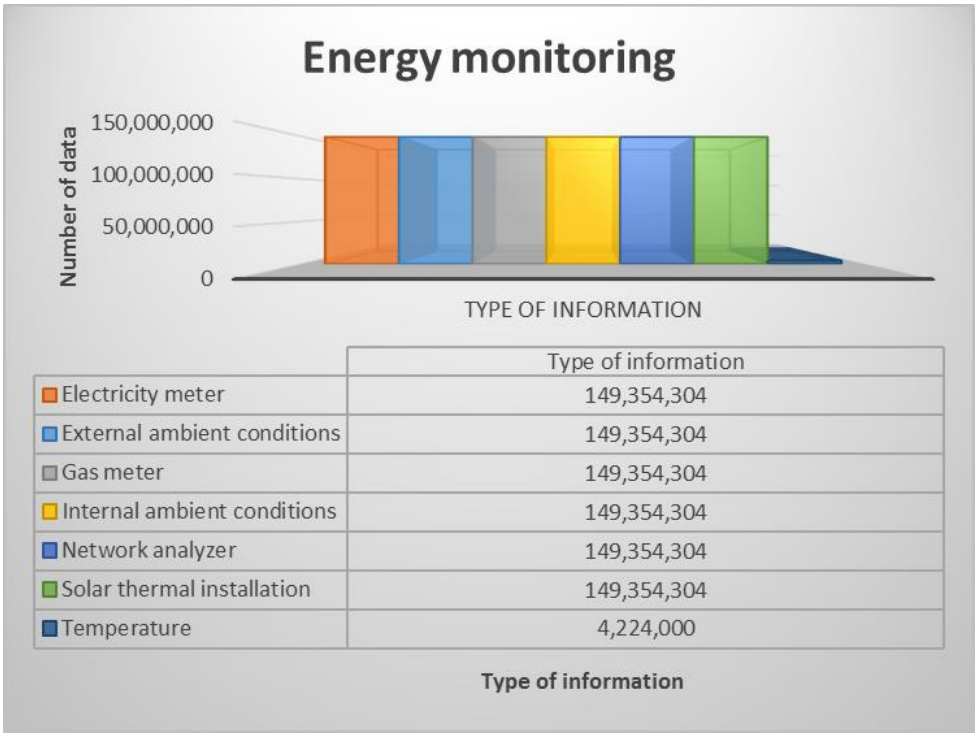


Figure 5.17Future Sending daily data in Energy monitoring category (Sentilo Platform)

- Noise monitoring category:

  The number of sensor (for each category) is as shown in below:

  - First type of noise: 10,000 sensors
  - Second type of noise: 10,000 sensors
  - Third type of noise: 10,000 sensors

  As shown in Table 5.11, the data production (by each sensor at each transaction) is shown in below:

  - First type of noise: 22 byte
  - Second type of noise: 22 byte
  - Third type of noise: 22 byte

  So, the total amount of the production data is 66 byte per transaction.

  As shown in Table 5.11, the data production (by each sensor per day) is shown in below:

  - First type of noise: 768 byte
  - Second type of noise: 31,680 byte
  - Third type of noise: 31,680 byte

  Therefore, the total amount of data is 65,472 byte which it must be transferred by each sensor at each transaction.

As shown in Figure 5.18, the total data production is listed in the below for each type of information in this category.

- First type of noise: 7,680,000 byte
- Second type of noise: 316,800,000 byte
- Third type of noise: 316,800,000 byte

As we shown in Table 5.11, the total amount of data production is 641,280,000 byte per day which it must be transferred to upper layer for further usage.

Table 5.11Data Volume Estimation in Barcelona Smart City

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Noise	10,000	22	768	7680000
	10,000	22	31,680	316,800,000
	10,000	22	31680	316800000

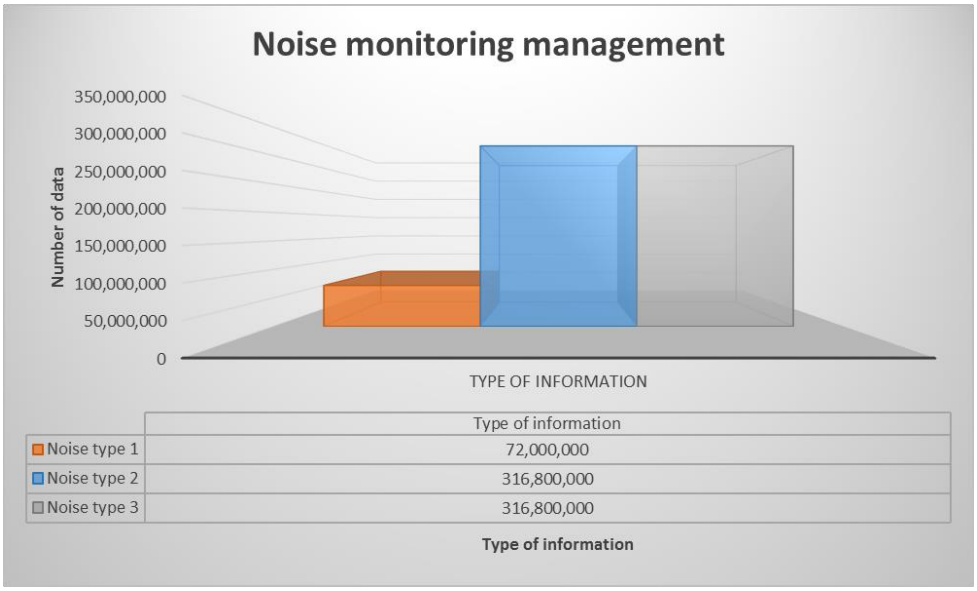


Figure 5.18Future Sending daily data in Noise management category (Sentilo Platform)

- Urban Lab monitoring management category:

The number of sensor (for each category) is as shown in below:

- Air quality: 40,000 sensors
- Bicycle flow: 40,000 sensors
- People flow: 40,000 sensors

- Traffic: 40,000 sensors
- Weather: 40,000 sensors

As shown in Table 5.12, the data production (by each sensor at each transaction) is shown in below:

- Air quality: 144 byte
- Bicycle flow: 22 byte
- People flow: 22 byte
- Traffic: 44 byte
- Weather: 120 byte

So, the total amount of the production data is 352 byte per transaction.

As shown in Table 5.12, the data production (by each sensor per day) is shown in below:

- Air quality: 13,824 byte
- Bicycle flow: 3,168 byte
- People flow: 3,168 byte
- Traffic: 63,360 byte
- Weather: 34,560 byte

Therefore, the total amount of data is 118,080 byte which it must be transferred by each sensor at each transaction.

As shown in Figure 5.19, the total data production is listed in the below for each type of information in this category.

- Air quality: 552,960,000 byte
- Bicycle flow: 126,720,000 byte
- People flow: 126,720,000 byte
- Traffic: 2,534,400,000 byte
- Weather: 1,382,400,000 byte

As we shown in Table 5.12, the total amount of data production is 4,723,200,000 byte per day which it must be transferred to upper layer for further usage.

Table 5.12Data Volume Estimation in Barcelona Smart City

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Air quality	40,000	144	13,824	552,960,000
Bicycle flow	40,000	22	3,168	126,720,000
People flow	40,000	22	3,168	126,720,000
Traffic	40,000	44	63,360	2,534,400,000
Weather	40,000	120	34,560	1,382,400,000
Total	200,000	352	118,080	4,723,200,000

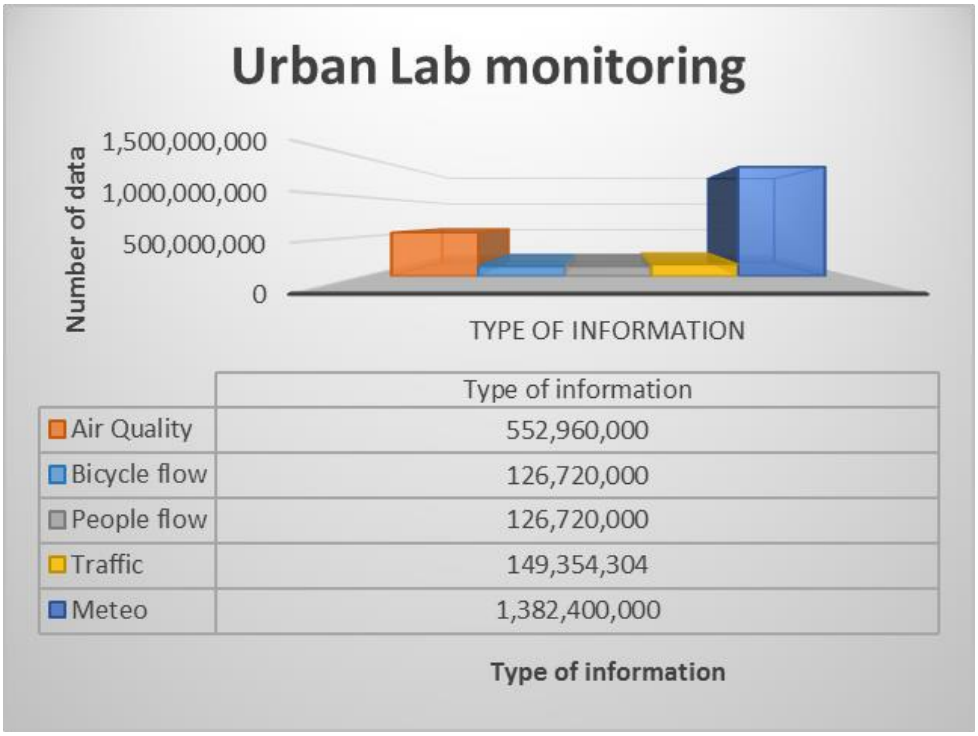


Figure 5.19Future Sending daily data in Urban Lab monitoring category (Sentilo Platform)

- Garbage Collection monitoring management

  The number of sensor (for each category) is as shown in below:

  - Glass container: 40,000 sensors
  - Organic container: 40,000 sensors
  - Paper container: 40,000 sensors

- Plastic container: 40,000 sensors
- Refuse container: 40,000 sensors

As shown in Table 5.13, the data production (by each sensor at each transaction) is shown in below:

- Glass container: 50 byte
- Organic container: 50 byte
- Paper container: 50 byte
- Plastic container: 50 byte
- Refuse container: 50 byte

So, the total amount of the production data is 250 byte per transaction.

As shown in Table 5.13, the data production (by each sensor per day) is shown in below:

- Glass container: 1,800 byte
- Organic container: 1,800 byte
- Paper container: 1,800 byte
- Plastic container: 1,800 byte
- Refuse container: 1,800 byte

Therefore, the total amount of data is 9,000 byte which it must be transferred by each sensor at each transaction.

As shown in Figure 5.20, the total data production is listed in the below for each type of information in this category.

- Glass container: 72,000,000 byte
- Organic container: 72,000,000 byte
- Paper container: 72,000,000 byte
- Plastic container: 72,000,000 byte
- Refuse container: 72,000,000 byte

As we shown in Table 5.13, the total amount of data production is 360,000,000 byte per day which it must be transferred to upper layer for further usage.

Table 5.13Data Volume Estimation in Barcelona Smart City

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Container glass	40,000	50	1,800	72,000,000
Container organic	40,000	50	1,800	72,000,000
Container paper	40,000	50	1,800	72,000,000
Container plastic	40,000	50	1,800	72,000,000
Container refuse	40,000	50	1,800	72,000,000
Total	200,000	250	9,000	360,000,000

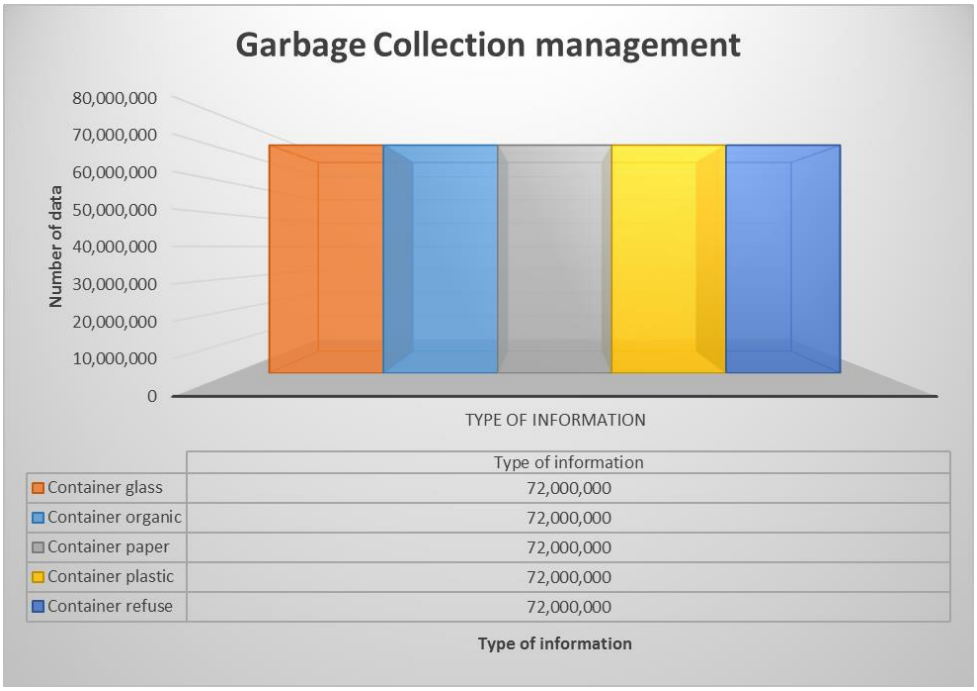


Figure 5.20Future Sending daily data in Garbage Collection management category (Sentilo Platform)

- Parking Spots monitoring management:

  The number of sensor (for each category) is as shown in below:

  - Parking: 80,000 sensors

As shown in Table 5.14, the data production (by each sensor at each transaction) is shown in below:

- Parking: 40 byte

So, the total amount of the production data is 40 byte per transaction.

As shown in Table 5.14, the data production (by each sensor per day) is shown in below:

- Parking: 4,000 byte

Therefore, the total amount of data is 4,000 byte which it must be transferred by each sensor at each transaction.

As shown in Figure 5.21, the total data production is listed in the below for each type of information in this category.

- Parking: 320,000,000 byte

As we shown in Table 5.14, the total amount of data production is 320,000,000 byte per day which it must be transferred to upper layer for further usage.

Table 5.14Data Volume Estimation in Barcelona Smart City

Type	Number of devices	Sending data (byte)		
		by each sensor at each transaction	by each sensor per day	Total amount of data per day
Parking	80,000	40	4,000	320,000,000
Total	80,000	40	4,000	320,000,000

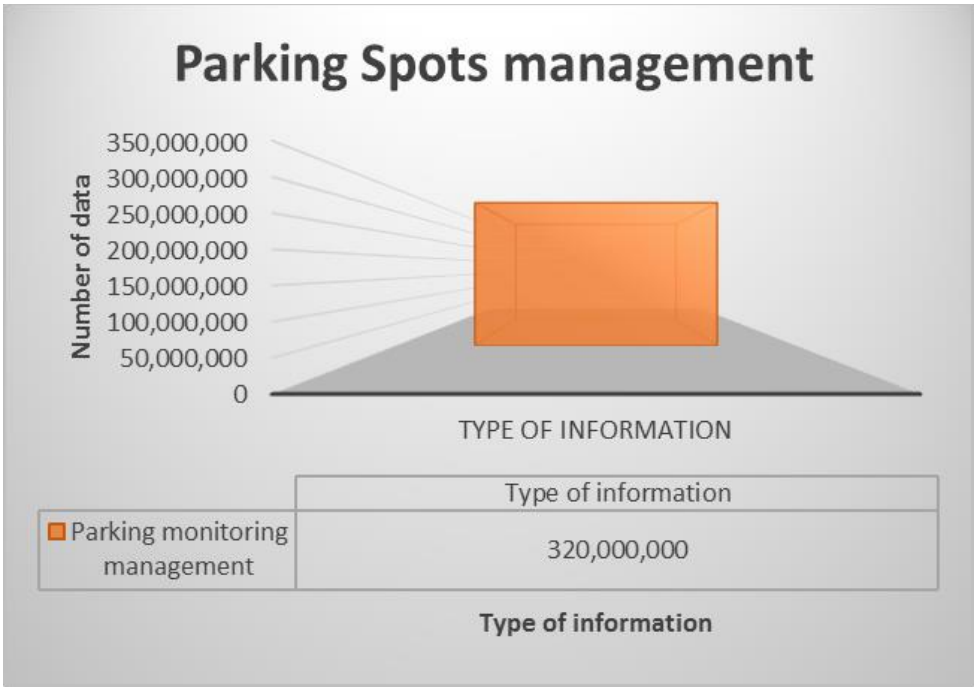


Figure 5.21Future Sending daily data in Parking Spots category (Sentilo Platform)

Next, we describe our results for the future data collection through those new data type in Barcelona city as shown in below:

- Water meter data (in Energy monitoring category):

  In Table 5.15 and Figure 5.22 and 5.23we show that each district of Barcelona has different number of sending data per transaction and per day as shown in below:

  - Cituat Vella has 40,105 households which produced almost 3,432,180 byte data per transaction.
  - Eixample has 40,105 households which produced almost 9,588,540 byte data per transaction.
  - Sans-Montjuic has has 73,682 households which produced almost 6,305,696 byte data per transaction.
  - Les Corts has 32,771 households which produced almost 2,804,538 byte data per transaction.
  - Sarria-Sant Gervasi has 56,113 households which produced almost 4,802,143 byte data per transaction.
  - Gracia has 52,478 households which produced almost 4,491,060 byte data per transaction.
  - Horta-Guinardi has 68,958 households which produced almost 5,901,417 byte data per transaction.
  - Nou-Barris has 66,008 households which produced almost 5,648,956 byte data per transaction.

- Sant Andreu has 58,836 households which produced almost 5,035,177 byte data per transaction.
- Sant Marti has 93,986 households which produced almost 8,043,310 byte data per transaction.

Indeed, the total amount of the produced data is almost 56,053,022 byte per transaction.

Table 5.15Water Meter Estimation in Barcelona Smart City

Type	City, Country		Number of house holds	Sending data per transaction (byte)	Sending data per transaction (byte) per day (every 15 minutes)	Sending data per transaction (MB) per day (every 15 minutes)
Water meter	Surrey, Canada		61,263	5,242,880		
	Barcelona, Spain					
	Districts	Total	654,979	56,053,022.21	5,381,090,131.95	5,381.09
		Ciutat Vella	40,105	3,432,180.96	329,489,372.55	329.49
		Eixample	112,042	9,588,540.57	920,499,894.75	920.50
		Sants-Montjuic	73,682	6,305,696.49	605,346,863.19	605.35
		Les Corts	32,771	2,804,538.15	269,235,662.08	269.24
		Sarrià-Sant Gervasi	56,113	4,802,143.63	461,005,788.85	461.01
		Gràcia	52,478	4,491,060.78	431,141,835.00	431.14
		Horta-Guinardó	68,958	5,901,417.15	566,536,046.68	566.54
		Nou Barris	66,008	5,648,956.52	542,299,825.54	542.30
		Sant Andreu	58,836	5,035,177.64	483,377,053.32	483.38
		Sant Martí	93,986	8,043,310.31	772,157,790.01	772.16

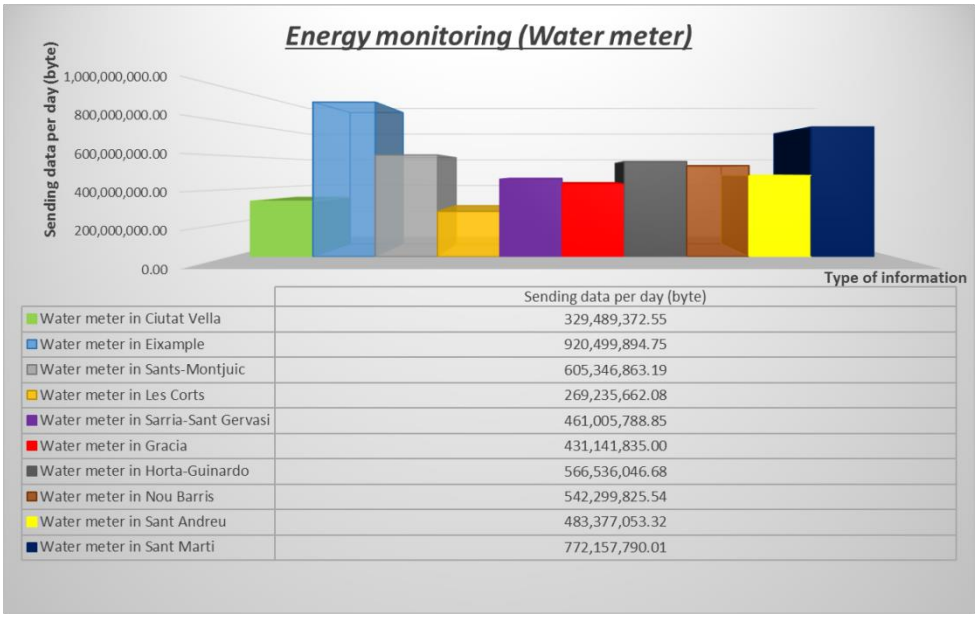


Figure 5.22Future Sending daily data in Water meter information (All Districts of Barcelona)

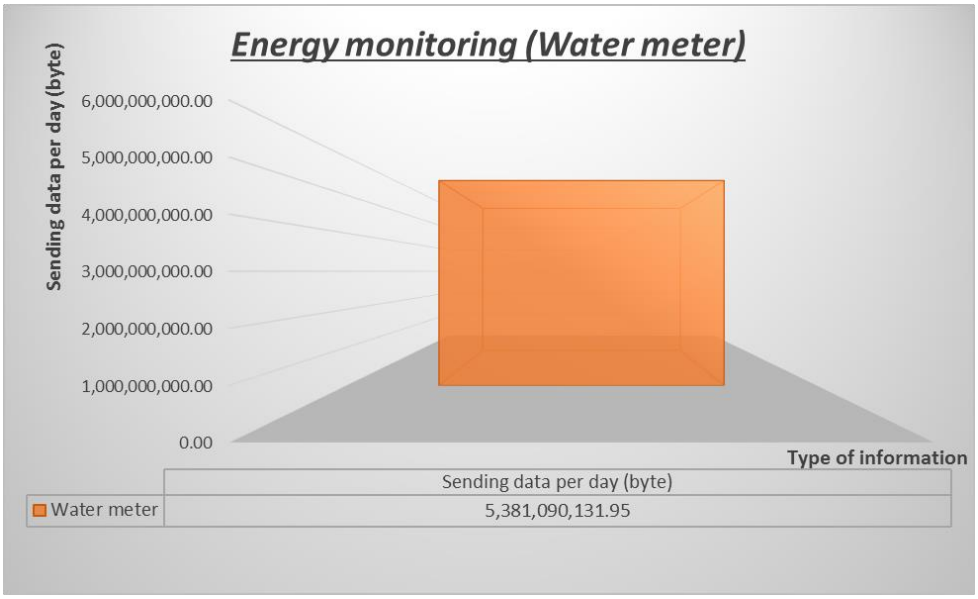


Figure 5.23Future Sending daily data in Water meter information

- Mobile application data:

  As depicted in Table 5.16 and Figure 5.24, we estimated that the total amount of the produced data (in three years) is 844,361,625,000 byte in the Barcelona Smart City of Barcelona (including,477,247,875,000 byte data for high-level range, 275,335,312,500 data for medium level-range and 91,778,437,500 byte data for low-level range). Similarly, almost 435 MB data will be produced for the high-level range in each transaction per day. And, almost 251 MB data will be generated for the medium-level range in each transaction per day. Finally, almost 83 MB data will be produced for low-level range in each transaction per day.

Table 5.16Mobile Application Estimation in Barcelona Smart City

Type	City, Country	Application	Number of active users	Total amount of data in 3 years		Total amount of data in 3 years (compress version)		Sending data per transaction per year		Sending data per transaction per day	
				MB	Byte	MB	Byte	MB	Byte	MB	Byte
Mobile APP	Barcelona, Spain	Sample	800	293.691	293,691,000.00	14	14,000,000.00	97.897	97897000	0.268210959	268210.9589
			Per user	0.36711375	367113.75	0.0175	17500	0.12	122,371.25	0.0003	335.26
		Coverage of users	More than 1 milion	477,247.88	477,247,875,000.00	22,750.00	22,750,000,000.00	159,082.63	159,082,625,000.00	435.84	435,842,808.22
			500,000 to 1 milion	275,335.31	275,335,312,500.00	13,125.00	13,125,000,000.00	91,778.44	91,778,437,500.00	251.45	251,447,773.97
			less than 500,000	91,778.44	91,778,437,500.00	4,375.00	4,375,000,000.00	30,592.81	30,592,812,500.00	83.82	83,815,924.66

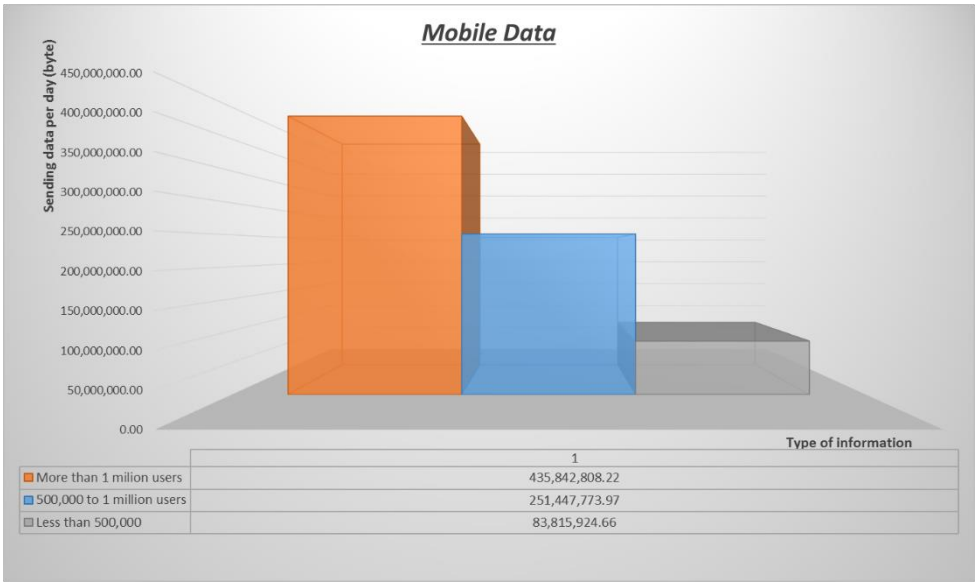


Figure 5.24Future Sending daily data in Mobile data information

- Camera Surveillance data: we measured that the total amount of produced data (per day) is 272,550,960,000,000 byte (including, 26,310,960,000,000 byte data in the airport, 172,800,000,000,000 byte data in the parking slots, and 73,440,000,000,000 byte data in the traffic lights) as shown in Figure 5.25 and Table 5.17.

Table 5.17Camera surveillance data Estimation in Barcelona Smart City

City, Country	Type of devices/equipments		Number of devices/equipments	Sending data per transaction (per hour/per camera)		Sending data per day (per hour/per camera)		Total amount of data per transaction		Total amount of data per day	
				MB	Byte	MB	Byte	MB	Byte	MB	Byte
Barcelona, Spain	Traffic lights		34,000	90	90,000,000	2,160	2,160,000,000	3,060,000	3,060,000,000,000	73,440,000	73,440,000,000,000
	Parking slots		80,000	90	90,000,000	2,160	2,160,000,000	7,200,000	7,200,000,000,000	172,800,000	172,800,000,000,000
	Airport (Terminal 1)	Check-in desks	166	90	90,000,000	2,160	2,160,000,000	14,940	14,940,000,000	358,560	358,560,000,000
		Baggage carousels	15	90	90,000,000	2,160	2,160,000,000	1,350	1,350,000,000	32,400	32,400,000,000
		Parking slots	12,000	90	90,000,000	2,160	2,160,000,000	1,080,000	1,080,000,000,000	25,920,000	25,920,000,000,000

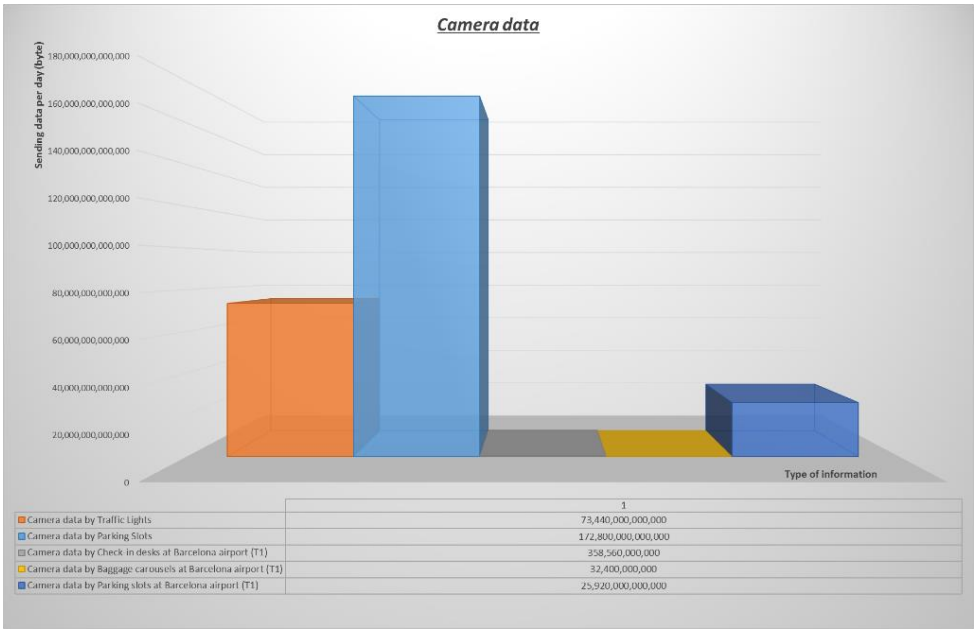


Figure 5.25Future Sending daily data in Camera Surveillance data information

- Vehicular Mobility data:

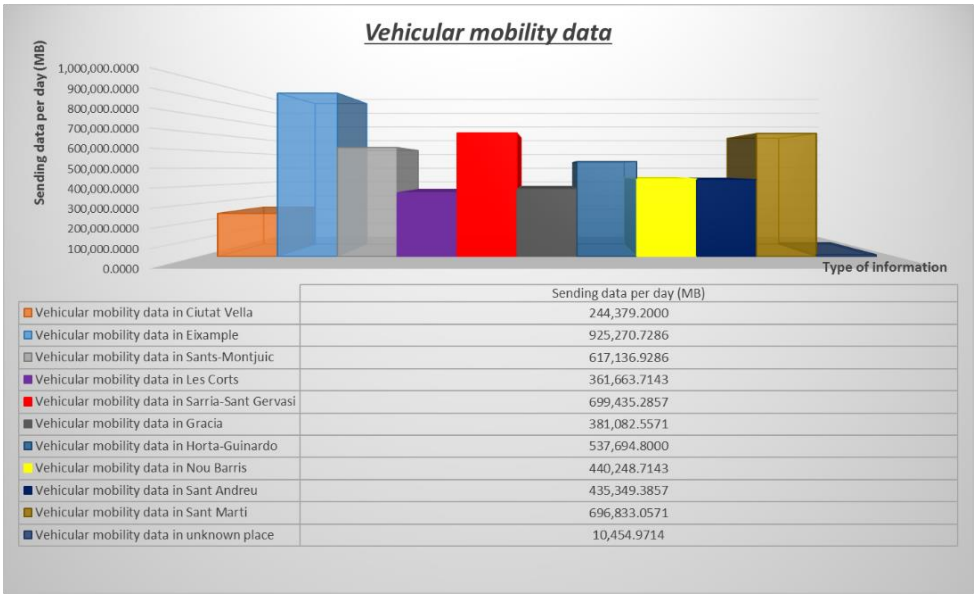
In Table 5.18 and Figure 5.26 and 5.27, we show that each district of Barcelona has different number of sending data per transaction and per day as shown in below:

- Cituat Vella has 42,448 registered cars which produced almost 244 GB data per day.
- Eixample has 160,717 registered cars which produced almost 925 GB data per day.
- Sants-Montjuic has 107,195 registered cars which produced almost 617 GB data per day.
- Les Corts has 62,820 registered cars which produced almost 361 GB data per day.
- Sarria-Sant Gervasi has 121,490 registered cars which produced almost 699 GB data per day.
- Gracia has 66,193 registered cars which produced almost 381 GB data per day.
- Horta-Guinardi has 92,706 registered cars which produced almost 537 GB data per day.
- Nou-Barris has 76,470 registered cars which produced almost 440 GB data per day.
- Sant Andreu has 75,619 registered cars which produced almost 435 GB data per day.
- Sant Marti has 121,038 registered cars which produced almost 696 GB data per day.
- Unknown location has 1,816 registered cars which produced almost 10 GB data per day.

Indeed, the total produced data is almost 5,345 GB data per day with 928,512 registered cars.

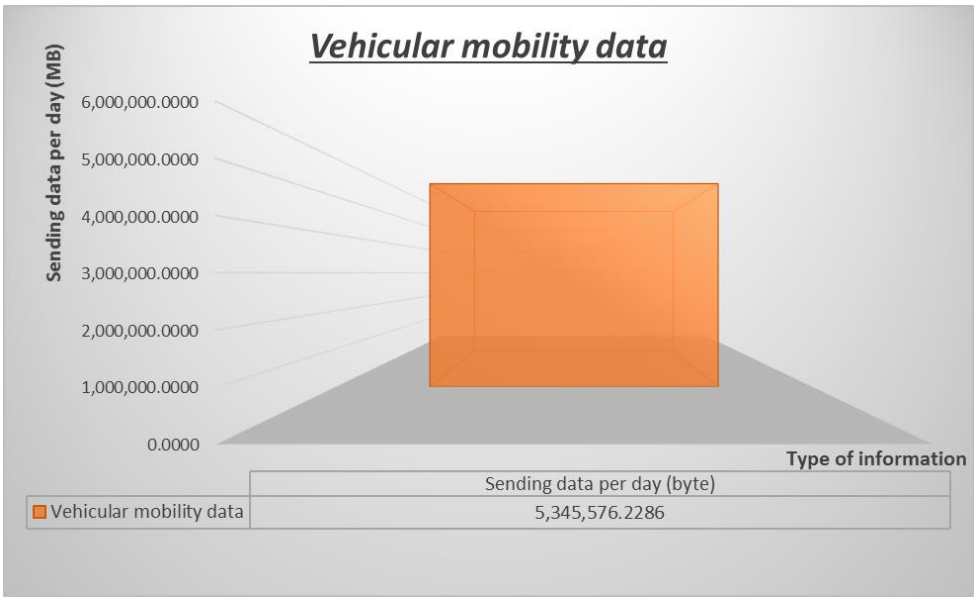
Table 5.18Vehicular Mobility data Estimation in Barcelona Smart City

City, Country		Number of cars	Sending data per day (GB)	Sending data per transaction (MB) per day	Sending data per transaction (GB) per year	Sending data per transaction (MB) per year
Koln, Germany		700	4.03	4030	1470.95	1470950
		1	0.0058	5.757142857	2.101357143	2151.789714
Barcelona, Spain						
Districts	Total	928,512	5,345.5762	5,345,576.2286	1,951,135.3234	1,951,135,323.4286
	Ciutat Vella	42,448	244.3792	244,379.2000	89,198.4080	89,198,408.0000
	Eixample	160,717	925.2707	925,270.7286	337,723.8159	337,723,815.9286
	Sants-Montjuïc	107,195	617.1369	617,136.9286	225,254.9789	225,254,978.9286
	Les Corts	62,820	361.6637	361,663.7143	132,007.2557	132,007,255.7143
	Sarrià-Sant Gervasi	121,490	699.4353	699,435.2857	255,293.8793	255,293,879.2857
	Gràcia	66,193	381.0826	381,082.5571	139,095.1334	139,095,133.3571
	Horta-Guinardó	92,706	537.6948	537,694.8000	196,258.6020	196,258,602.0000
	Nou Barris	76,470	440.2487	440,248.7143	160,690.7807	160,690,780.7143
	Sant Andreu	75,619	435.3494	435,349.3857	158,902.5258	158,902,525.7857
	Sant Martí	121,038	696.8331	696,833.0571	254,344.0659	254,344,065.8571
	Does not appear	1,816	10.4550	10,454.9714	3,816.0646	3,816,064.5714



	Sending data per day (MB)
Vehicular mobility data in Ciutat Vella	244,379.2000
Vehicular mobility data in Eixample	925,270.7286
Vehicular mobility data in Sants-Montjuic	617,136.9286
Vehicular mobility data in Les Corts	361,663.7143
Vehicular mobility data in Sarria-Sant Gervasi	699,435.2857
Vehicular mobility data in Gracia	381,082.5571
Vehicular mobility data in Horta-Guinardo	537,694.8000
Vehicular mobility data in Nou Barris	440,248.7143
Vehicular mobility data in Sant Andreu	435,349.3857
Vehicular mobility data in Sant Marti	696,833.0571
Vehicular mobility data in unknown place	10,454.9714

Figure 5.26Future Sending daily data in Vehicular Mobility information (All Districts)



	Sending data per day (byte)
Vehicular mobility data	5,345,576.2286

Figure 5.27Future data byVehicular Mobilitydata type

5.3.5 Data Filtering Measurements in Barcelona

In this section, we aim to present how data filtering can be applied in real Smart City. So, we describe data aggregation and data compression in this example. As we know, we have the different type of data aggregation and data compression. Here we just say some sample of data aggregation and data compression which we are able to depict in our F2C model through Barcelona Smart City as shown in below sub sections.

As we discussed in last sections, data aggregation is kind of process to optimize collected data with different techniques. Plus, data compression is able to provide more facility for reducing the size of data in transfer time. The F2C data model provides numerous facility to apply data aggregation and data compression techniques and algorithms in Smart City. Now, we present how

we can apply some basic aggregation and compression techniques as an example of optimization smodels in Smart City.

- Redundant data elimination technique: In this technique, we focus on providing a basic yet effective solution to easily reduce the amount of duplicated data collected from the sensors layer. In fact, there is the number of redundant data which are produced by sensors every moment. For example, in the case of weather measurement, each sensor sends the current temperature measurements, but this type of data is prone to repetitions, so eliminating them may easily reduce such amount of data.
- Compression: As data is collected and transmitted to an upper level delayed, there are some options to accumulate a reasonable amount of data and compute compression, in order to obviously reduce the amount of data transfer.

5.3.5.1 Methodology

Here we aim to describe our research methodology for our proposed data aggregation and data compression techniques in our F2C model as shown more details in below:

1- Depicting our scenario to Smart City of Barcelona:

According to the current distribution of districts and sections in Barcelona, we estimate that our Fog-layer-1 can be covered with 73 Fog-Area, which is matched with the number of sections in Barcelona. In this case, our Fog-Area covers almost 1 km2, which is a reasonable fog node size. In addition, the Fog-Leader can be defined as 10 main nodes which are matched with the number of the district in Barcelona. As shown in Figure5.28, the figure shows how the section and districts can be fixed in our F2C model with respect to data aggregation model.

In our proposed model, data aggregation and compression can be applied in any layer of fog to cloud layer as shown in Figure 5.27 as shown more details in below:

- Fog-Leader = 10; (each district of Barcelona has three sections. By other words, each Fog-Leader is responsible for the different number of Fog-Devices. For example, regarding the official report by city hall of Barcelona, Ciutat Vella district covers with 4 Fog-Aeras. However, Eixample includes with 6 Fog-Areas)
- Fog-Area=73; (each section of Barcelona has three Fog-Devices. By other words, we assumed that each Fog-Area has three different number of Fog-Devices)
- Fog-Device=219; (we assumed that each Edge-Data-Source has the same number of edge-data-sources)
- Edge-Data-Sources=number of sensor devices.

Therefore, for instance, Ciutat Vella district covers with four sections (Fog-Area), twelve Fog-Device, and almost 969 sensors devices (Edge-data-sources), in particular forthe electricity meter information under energy monitoring category.

2- The Different level of aggregation and compression in Smart City of Barcelona:

With respect to our description scenario and the number of districts and sections in Barcelona, we assume that we have three different level of aggregation and compression from edge to cloud in the city of Barcelona. Therefore, in the Fog-layer-1 we can have aggregation in Fog-Device, Fog-Leader (Section), and Fog-Leader (District). Therefore, as shown in Figure 5.28, we apply four level of aggregation and compression in our model. However, we have the possibility to reduce the number aggregation and compression in our model. Plus, there is another possibility to have one more layer of data aggregation in the cloud.

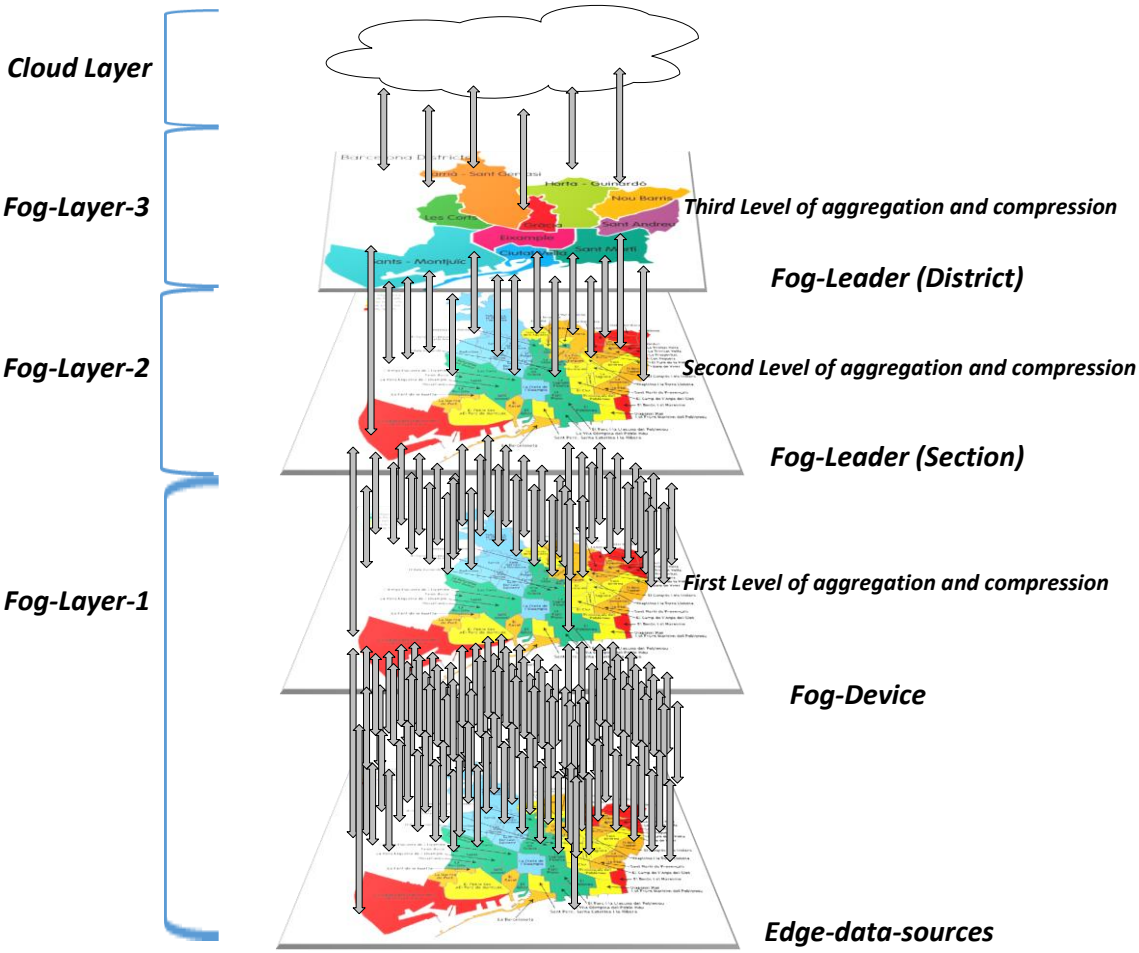


Figure 5.28Level of Aggregation and Compression in Smart City of Barcelona

3- Redundant data rate in Sentilo platform

We calculated the redundant data rate in Sentilo platform. So, we monitored the content of produced data in one day. So, we explored that the redundant data rate for each category of information is different. As we observed, the redundant data for energy, noise, garbage, parking and urban is around 50%, 75%, 70%, 40%, and 30% respectively. For example, as you can see in Table5.19, we monitored garbage collection data category in Sentilo

platform. So, as you can see in this table there is almost 70% redundant data in this category per day.

Table 5.19Redundant Data in Garbage Collection Monitoring

Type of information		Container of Glass	Container of Organic	Container of Paper	Container of Plastic	Container of Refuse
Number of devices		57	71	57	205	277
Period of sending information	1:00 AM	65%	55%	68%	85%	88%
	2:00 AM	65%	75%	72%	55%	80%
	3:00 AM	71%	64%	89%	74%	82%
	4:00 AM	62%	72%	79%	62%	69%
	5:00 AM	75%	70%	62%	71%	78%
	6:00 AM	60%	50%	76%	61%	80%
	7:00 AM	60%	70%	69%	74%	82%
	8:00 AM	80%	85%	68%	63%	75%
	9:00 AM	72%	62%	75%	56%	72%
	10:00 AM	74%	71%	80%	66%	81%
	11:00 AM	74%	79%	66%	71%	69%
	12:00 PM	60%	50%	77%	56%	69%
	1:00 PM	80%	70%	79%	68%	73%
	2:00 PM	70%	71%	83%	69%	81%
	3:00 PM	71%	78%	84%	56%	78%
	4:00 PM	74%	64%	84%	69%	66%
	5:00 PM	66%	62%	62%	81%	67%
	6:00 PM	65%	75%	88%	44%	75%
	7:00 PM	72%	70%	77%	77%	72%
	8:00 PM	78%	68%	70%	57%	69%
	9:00 PM	63%	73%	76%	81%	77%
	10:00 PM	62%	72%	81%	66%	82%
	11:00 PM	70%	80%	77%	70%	79%
	12:00 AM	74%	84%	79%	79%	81%
Total number in each category		69%	70%	76%	67%	76%
Total number in all category		72%				

4- Compression rate

We can apply the data compression technique after using data aggregation techniques in order to further reduce the amounts of data to be transferred to higher layers. The Zip format, for instance, is one solution provided by PKWARE in 1989 [205]. In this experiment, we have used the Zip format in our model to perform compression at all layer.

5- Estimation of Optimization data through Data Aggregation and Data Compression models:

The data classification phase classifies and organizes all data collected from the different categories of sensors. In our use case, Sentilo provides five categories of information and services which are energy, noise, urban, garbage, and parking. Each category is divided into different types of information. For instance, the energy category contains electricity meter, external ambient conditions, gas meter, internal ambient conditions, network analyzer, solar thermal installation, and temperature. The noise category includes has three different types of information. The urban category encompasses to air quality, bicycle flow, people flow, traffic and weather. The garbage category has container glass, container organic, container paper, container plastic, and container refuse. And finally, the parking category has only one type of information.

The two basic data aggregation techniques explored will be implemented in all layer (including Edge-data-source, Fog-Device, Fog-Leader (in Fog-Layer-2), and Fog-Leader (in Fog-Layer-3)) as explained before. So, in the previous section, we calculate the number of produced data in each category. And then, regarding the redundant data rate for each category of information, we can calculate the total number of produced data after data aggregation and data compression in each layer as shown in Figure 5.28. Later, we will discuss more data aggregation and compression rate.

5.3.5.2Results

As shown in previous sections, we calculated that the Sentilo generated different amount of data per day for different categories of information (including, energy monitoring, noise monitoring, garbage collection, parking spot monitoring, and urban lab monitoring) as listed in below:

- Energy monitoring category: 2,390,344,704 byte per day.
- Noise monitoring category: 641,280,000 byte per day.
- Garbage Collection category: 360,000,000 byte per day.
- Parking Spot monitoring category: 320,000,000 byte per day.
- Urban Lab monitoring category: 4,723,200,000 byte per day.

We realized that each category of information produced the huge amount of the redundant data in every transaction (by different sensors) and per day. Therefore, we monitored a single day of data generation in Sentilo platform. Then, we observed that the redundant data for energy, noise, garbage, parking and urban is almost 50%, 75%, 70%, 40%, and 30% respectively. Therefore, we have an initial thought to absorb this amount of data through the layers of our F2C data management architecture.

As we mentioned in the previous section, we applied two different techniques to reduce the number of data transfer among F2C layers. First, we used data aggregation techniques to eliminate the number of redundant data in the layer. Second, we used data compression techniques (for example, one solution proposed by PKWARE [205]) to compress data size after applying aggregation techniques. Note that data aggregation and data compression techniques can be implemented in each layer of F2C data architecture. Therefore, in the following paragraph, we will explain how much data will be reduced at each layer.

The fog device is in the first level of our aggregation model. With respect to number of the redundant observation, we calculated that sensors data would be reduced to 1,194,834,432 bytes for energy monitoring, 160,320,000 bytes for noise monitoring, 108,000,000 bytes for garbage collection, 192,000,000 bytes for parking spot monitoring, and 3,306,240,000 bytes for urban lab as shown in Table 5.20 and Figure 29(blue lines). Then, the amount of data can be further decreased to smaller sizes through data compression. Therefore, the data size will be 262,863,575 bytes for the energy monitoring, 35,270,400 bytes for the noise monitoring, 23,760,000 bytes for the garbage collection, 42,240,000 bytes for the parking, and 727,372,800 bytes for the urban lab as shown in Table 5.20 and Figure 29(green lines).

Similarly, the fog leaders at Fog-Layer-2 (city sections) play the second level for performing data aggregation and data compression techniques. Therefore, the data volume will be reduced to 597,586,176 bytes for energy monitoring, 39,498,840 bytes for noise monitoring, 32,462,370 bytes for garbage collection, 115,106,400 bytes for parking spot monitoring, and 2,318,823,158 bytes for the urban lab as shown in Table 5.20 and Figure 29(blue lines). Then, the number of data can be shifted to smaller sizes through data compression. Therefore, the data size will go to 131,468,959 bytes for the energy monitoring, 8,689,745 bytes for the noise monitoring, 7,141,721 bytes for garbage collection, 25,323,408 bytes for the parking, and 510,141,095 bytes for the urban lab as shown in Table 5.20 and Figure 29(green lines).

Next layer is the fog leaders at Fog-Layer-3 (city districts) to handle data compression and data compression techniques. In this layer, data size goes to 298,793,088 bytes for energy monitoring, 9,874,710 bytes for noise monitoring, 9,738,711 bytes for garbage collection, 69,063,840 bytes for parking spot monitoring, and 1,623,176,211 bytes for urban lab after handling data aggregation as shown in Table 5.20 and Figure 29(blue lines). Then, the number of data can be further reduced through data compression. Therefore, the data size will change to 65,734,479 bytes for the energy monitoring, 2,172,436 bytes for the noise monitoring, 2,142,516 bytes for garbage collection, 15,194,045 bytes for the parking, and 35,098,766 bytes for the urban lab as shown in Table 5.20 and Figure 29 (green lines).

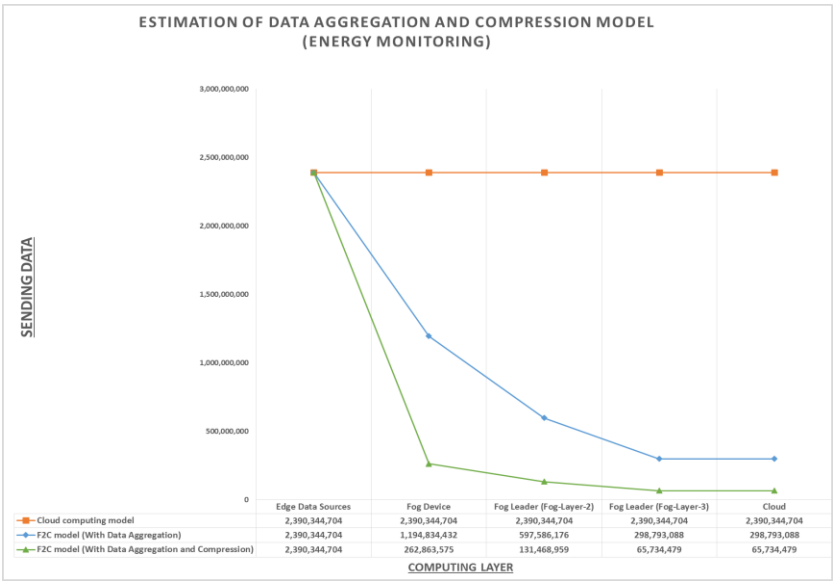
After these computations, we conclude that: i) aggregation efficiency rate at fog devices, fog leaders in Fog-Layer-2 (city sections), and fog leader in Fog-Layer-3 (city districts): is almost 49.98%, 50.01%, and 50.08% efficiency rate for energy monitoring information. Similarly, the noise monitoring efficiency rate is 24.96%, 25.02%, and 25.05%. Then, the garbage collection rate is 29.99%, 30.05%, and 30.08%. In addition, the parking spot monitoring rate is 59.95%, 59.99%, and 60.01%. Indeed, the urban lab-monitoring rate is 68.93%, 69.13%, and 70.01%. ii) Compression efficiency rate is almost 22% for all layers (including Fog device, Fog-Leader (in the Fog-Layer-2 layer), and Fog-Leader (in the Fog-Layer-3 layer)).

Indeed, the total efficiency rate (including redundant data elimination and compression) at fog devices, fog leaders in Fog-Layer-2 (city sections), and fog leader in Fog-Layer-3 (city districts) is as shown in below:

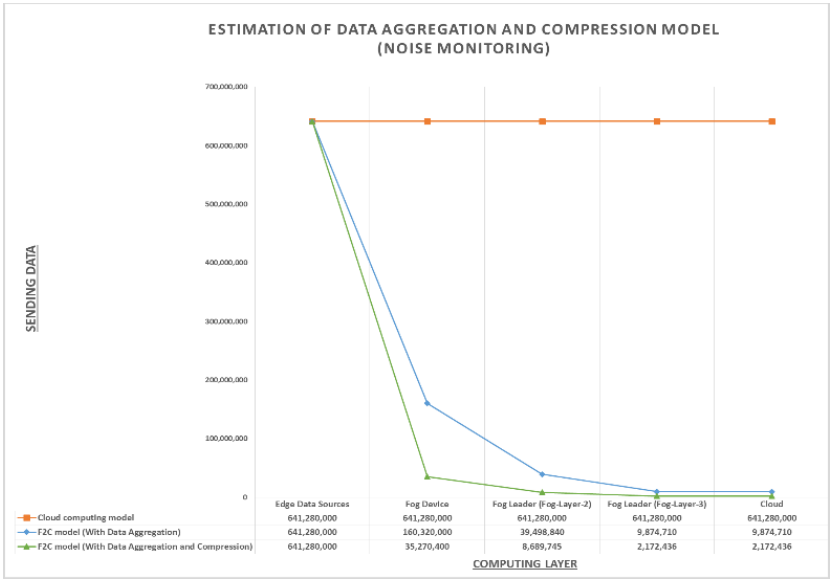
- Energy Monitoring category: About 71.98%, 72.01%, and 72.08%.
- Noise Monitoring category: Almost 46.96%, 47.02%, and 47.05%.
- Garbage Collection category: Approximately 51.99%, 52.05%, and 52.08%.
- Parking Spot Monitoring category: Almost 81.95%, 81.99%, and 82.01%.
- Urban Lab Monitoring category: About 90.93%, 91.13%, and 92.01%.

Table 5.20Redundant Data Aggregation Model

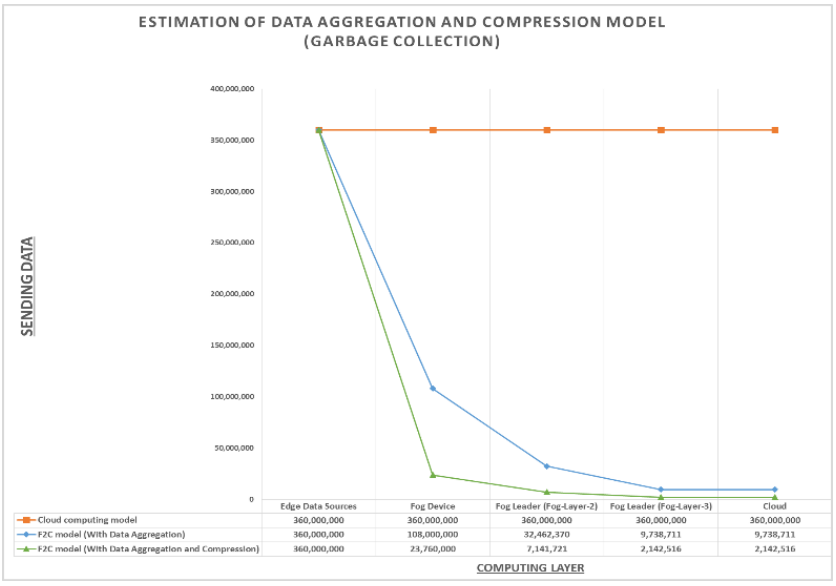
Category of inforamtion	Type of information	Number of sensors	Computing model	Data Filtering techniques	Sections of Barcelona				Districts of Barcelona										Total Number of Fog Layer 3
					All				Ciutat Vella	Eixample	Sants-Montjuic	Les Corts	Sarria-Sant Gervasi	Gracia	Horta-Guinardo	Nou Barris	Sant Andreu	Sant Marti	
					Fog Layer 1			Fog Layer 2	Fog Layer 3										
					Edge Data Sources	Fog Device	Fog-Leader	Fog -Leader	Fog-Leader										
Energy monitoring	Electricity meter	70,717	Cloud Computing	No Action	2,112	2,112	682,176	2,046,528	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
					149,354,304	149,354,304	149,396,544	149,396,544	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
			Fog Computing	Data Aggregation	2,112	1,056	170,544	255,816	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
					149,354,304	74,677,152	37,349,136	18,674,568	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
	External ambient conditions	70,717	Cloud Computing	No Action	2,112	2,112	682,176	2,046,528	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
					149,354,304	149,354,304	149,396,544	149,396,544	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
			Fog Computing	Data Aggregation	2,112	1,056	170,544	255,816	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
					149,354,304	74,677,152	37,349,136	18,674,568	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
	Internal ambient conditions	70,717	Cloud Computing	No Action	2,112	2,112	682,176	2,046,528	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
					149,354,304	149,354,304	149,396,544	149,396,544	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
			Fog Computing	Data Aggregation	2,112	1,056	170,544	255,816	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
					149,354,304	74,677,152	37,349,136	18,674,568	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
	Network analyzer	70,717	Cloud Computing	No Action	23,232	23,232	7,503,936	22,511,808	90,047,232	135,070,848	180,094,464	67,535,424	135,070,848	112,559,040	247,629,888	292,653,504	157,582,656	225,118,080	1,643,361,984
					1,642,897,344	1,642,897,344	1,643,361,984	1,643,361,984	90,047,232	135,070,848	180,094,464	67,535,424	135,070,848	112,559,040	247,629,888	292,653,504	157,582,656	225,118,080	1,643,361,984
			Fog Computing	Data Aggregation	23,232	11,616	1,875,984	2,813,976	11,255,904	16,883,856	22,511,808	8,441,928	16,883,856	14,069,880	30,953,736	36,581,688	19,697,832	28,139,760	205,420,248
					1,642,897,344	821,448,672	410,840,496	205,420,248	11,255,904	16,883,856	22,511,808	8,441,928	16,883,856	14,069,880	30,953,736	36,581,688	19,697,832	28,139,760	205,420,248
	Solar thermal installation	70,717	Cloud Computing	No Action	2,112	2,112	682,176	2,046,528	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
					149,354,304	149,354,304	149,396,544	149,396,544	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
			Fog Computing	Data Aggregation	2,112	1,056	170,544	255,816	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
					149,354,304	74,677,152	37,349,136	18,674,568	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
	Temperature	70,717	Cloud Computing	No Action	2,112	2,112	682,176	2,046,528	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
					149,354,304	149,354,304	149,396,544	149,396,544	8,186,112	12,279,168	16,372,224	6,139,584	12,279,168	10,232,640	22,511,808	26,604,864	14,325,696	20,465,280	149,396,544
			Fog Computing	Data Aggregation	2,112	1,056	170,544	255,816	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
					149,354,304	74,677,152	37,349,136	18,674,568	1,023,264	1,534,896	2,046,528	767,448	1,534,896	1,279,080	2,813,976	3,325,608	1,790,712	2,558,160	18,674,568
	Total number	424,302	Cloud Computing	No Action	2,390,344,704	2,390,344,704	2,390,344,704	2,390,344,704	2,390,344,704										
			Fog Computing	Data Aggregation	2,390,344,704	1,194,834,432	597,586,176	298,793,088	298,793,088										
	Total number	424,302	Cloud Computing	No Action	2,390,344,704	2,390,344,704	2,390,344,704	2,390,344,704	2,390,344,704										
			Fog Computing	Data Aggregation and Data Compression	2,390,344,704	262,863,575	131,468,959	65,734,479	65,734,479										
Noise monitoring	Noise	10,000	Cloud Computing	No Action	768	768	34,560	103,680	414,720	622,080	829,440	311,040	622,080	518,400	1,140,480	1,347,840	725,760	1,036,800	7,680,000
					7,680,000	7,680,000	7,680,000	7,680,000	414,720	622,080	829,440	311,040	622,080	518,400	1,140,480	1,347,840	725,760	1,036,800	7,680,000
			Fog Computing	Data Aggregation	768	192	2,160	1,620	6,480	9,720	12,960	4,860	9,720	8,100	17,820	21,060	11,340	16,200	118,260
					7,680,000	1,920,000	473,040	118,260	6,480	9,720	12,960	4,860	9,720	8,100	17,820	21,060	11,340	16,200	118,260
		10,000	Cloud Computing	No Action	31,680	31,680	1,425,600	4,276,800	17,107,200	25,660,800	34,214,400	12,830,400	25,660,800	21,384,000	47,044,800	55,598,400	29,937,600	42,768,000	316,800,000
					316,800,000	316,800,000	316,800,000	316,800,000	17,107,200	25,660,800	34,214,400	12,830,400	25,660,800	21,384,000	47,044,800	55,598,400	29,937,600	42,768,000	316,800,000
			Fog Computing	Data Aggregation	31,680	7,920	89,100	66,825	267,300	400,950	534,600	200,475	400,950	334,125	735,075	868,725	467,775	668,250	4,878,225
					316,800,000	79,200,000	19,512,900	4,878,225	267,300	400,950	534,600	200,475	400,950	334,125	735,075	868,725	467,775	668,250	4,878,225
		10,000	Cloud Computing	No Action	31,680	31,680	1,425,600	4,276,800	17,107,200	25,660,800	34,214,400	12,830,400	25,660,800	21,384,000	47,044,800	55,598,400	29,937,600	42,768,000	316,800,000
					316,800,000	316,800,000	316,800,000	316,800,000	17,107,200	25,660,800	34,214,400	12,830,400	25,660,800	21,384,000	47,044,800	55,598,400	29,937,600	42,768,000	316,800,000
			Fog Computing	Data Aggregation	31,680	7,920	89,100	66,825	267,300	400,950	534,600	200,475	400,950	334,125	735,075	868,725	467,775	668,250	4,878,225
					316,800,000	79,200,000	19,512,900	4,878,225	267,300	400,950	534,600	200,475	400,950	334,125	735,075	868,725	467,775	668,250	4,878,225
	Total number	424,302	Cloud Computing	No Action	641,280,000	641,280,000	641,280,000	641,280,000	641,280,000										
			Fog Computing	Data Aggregation	641,280,000	160,320,000	39,498,840	9,874,710	9,874,710										
	Total number	424,302	Cloud Computing	No Action	641,280,000	641,280,000	641,280,000	641,280,000	641,280,000										
			Fog Computing	Data Aggregation and Data Compression	641,280,000	35,270,400	8,689,745	2,172,436	2,172,436										
Garbage Collection	Container glass	40,000	Cloud Computing	No Action	1,800	1,800	329,400	988,200	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
					72,000,000	72,000,000	72,000,000	72,000,000	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
			Fog Computing	Data Aggregation	1,800	540	29,646	26,681	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
					72,000,000	21,600,000	6,492,474	1,947,742	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
	Container organic	40,000	Cloud Computing	No Action	1,800	1,800	329,400	988,200	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
					72,000,000	72,000,000	72,000,000	72,000,000	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
			Fog Computing	Data Aggregation	1,800	540	29,646	26,681	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
					72,000,000	21,600,000	6,492,474	1,947,742	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
	Container paper	40,000	Cloud Computing	No Action	1,800	1,800	329,400	988,200	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
					72,000,000	72,000,000	72,000,000	72,000,000	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
			Fog Computing	Data Aggregation	1,800	540	29,646	26,681	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
					72,000,000	21,600,000	6,492,474	1,947,742	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
	Container plastic	40,000	Cloud Computing	No Action	1,800	1,800	329,400	988,200	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
					72,000,000	72,000,000	72,000,000	72,000,000	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
			Fog Computing	Data Aggregation	1,800	540	29,646	26,681	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
					72,000,000	21,600,000	6,492,474	1,947,742	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
	Container refuse	40,000	Cloud Computing	No Action	1,800	1,800	329,400	988,200	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
					72,000,000	72,000,000	72,000,000	72,000,000	3,952,800	5,929,200	7,905,600	2,964,600	5,929,200	4,941,000	10,870,200	12,846,600	6,917,400	9,882,000	72,000,000
			Fog Computing	Data Aggregation	1,800	540	29,646	26,681	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
					72,000,000	21,600,000	6,492,474	1,947,742	106,726	160,088	213,451	80,044	160,088	133,407	293,495	346,858	186,770	266,814	1,947,742
	Total number	424,302	Cloud Computing	No Action	360,000,000	360,000,000	360,000,000	360,000,000	360,000,000										
			Fog Computing	Data Aggregation	360,000,000	108,000,000	32,462,370	9,738,711	9,738,711										
	Total number	200,000	Cloud Computing	No Action	360,000,000	360,000,000	360,000,000	360,000,000	360,000,000										
			Fog Computing	Data Aggregation and Data Compression	360,000,000	23,760,000	7,141,721	2,142,516	2,142,516										
Parking Spot	Parking	80,000	Cloud Computing	No Action	4,000	4,000	1,460,000	4,380,000	17,520,000	26,280,000	35,040,000	13,140,000	26,280,000	21,900,000	48,180,000	56,940,000	30,660,000	43,800,000	320,000,000
					320,000,000	320,000,000	320,000,000	320,000,000	17,520,000	26,280,000	35,040,000	13,140,000	26,280,000	21,900,000	48,180,000	56,940,000	30,660,000	43,800,000	320,000,000
			Fog Computing	Data Aggregation	1,800	2,400	525,600	946,080	3,784,320	5,676,480	7,568,640	2,838,240	5,676,480	4,730,400	10,406,880	12,299,040	6,622,560	9,460,800	69,063,840
					320,000,000	192,000,000	115,106,400	69,063,840	3,784,320	5,676,480	7,568,640	2,838,240	5,676,480	4,730,400	10,406,880	12,299,040	6,622,560	9,460,800	69,063,840
	Total number	200,000	Cloud Computing	No Action	320,000,000	320,000,000	320,000,000	320,000,000	320,000,000										
			Fog Computing	Data Aggregation	320,000,000	192,000,000	115,106,400	69,063,840	69,063,840										
	Total number	80,000	Cloud Computing	No Action	320,000,000	320,000,000	320,000,000	320,000,000	320,000,000										
			Fog Computing	Data Aggregation and Data Compression	320,000,000	42,240,000	25,323,408	15,194,045	15,194,045										
Urban Lab monitoring	Air quality	40,000	Cloud Computing	No Action	13,824	13,824	2,529,792	7,589,376	30,357,504	45,536,256	60,715,008	22,768,128	45,536,256	37,946,880	83,483,136	98,661,888	53,125,632	75,893,760	552,960,000
					552,960,000	552,960,000	552,960,000	552,960,000	30,357,504	45,536,256	60,715,008	22,768,128	45,536,256	37,946,880	83,483,136	98,661,888	53,125,632	75,893,760	552,960,000
			Fog Computing	Data Aggregation	13,824	9,677	1,239,598	2,603,156	10,412,624	15,618,936	20,825,248	7,809,468	15,618,936	13,015,780	28,634,716	33,841,028	18,222,092	26,031,560	190,030,386
					552,960,000	387,072,000	271,471,980	190,030,386	10,412,624	15,618,936	20,825,248	7,809,468	15,618,936	13,015,780	28,634,716	33,841,028	18,222,092	26,031,560	190,030,386
	Bicycle flow	40,000	Cloud Computing	No Action	3,168	3,168	579,744	1,739,232	6,956,928	10,435,392	13,913,856	5,217,696	10,435,392	8,696,160	19,131,552	22,610,016	12,174,624	17,392,320	126,720,000
					126,720,000	126,720,000	126,720,000	126,720,000	6,956,928	10,435,392	13,913,856	5,217,696	10,435,392	8,696,160	19,131,552	22,610,016	12,174,624	17,392,320	126,720,000
			Fog Computing	Data Aggregation	3,168	2,218	284,075	596,557	2,386,226	3,579,339	4,772,453	1,789,670	3,579,339	2,982,783	6,562,122	7,755,235	4,175,896	5,965,566	43,548,630
					126,720,000	88,704,000	62,212,329	43,548,630	2,386,226	3,579,339	4,772,453	1,789,670	3,579,339	2,982,783	6,562,122	7,755,235	4,175,896	5,965,566	43,548,630
	People flow	40,000	Cloud Computing	No Action	3,168	3,168	579,744	1,739,232	6,956,928	10,435,392	13,913,856	5,217,696	10,435,392	8,696,160	19,131,552	22,610,016	12,174,624	17,392,320	126,720,000
					126,720,000	126,720,000	126,720,000	126,720,000	6,956,928	10,435,392	13,913,856	5,217,696	10,435,392	8,696,160	19,131,552	22,610,016	12,174,624	17,392,320	126,720,000
			Fog Computing	Data Aggregation	3,168	2,218	284,075	596,557	2,386,226	3,579,339	4,772,453	1,789,670	3,579,339	2,982,783	6,562,122	7,755,235	4,175,896	5,965,566	43,548,630
					126,720,000	88,704,000	62,212,329	43,548,630	2,386,226	3,579,339	4,772,453	1,789,670	3,579,339	2,982,783	6,562,122	7,755,235	4,175,896	5,965,566	43,548,630
	Traffic	40,000	Cloud Computing	No Action	63,360	63,360	11,594,880	34,784,640	139,138,560	208,707,840	278,277,120	104,353,920	208,707,840	173,923,200	382,631,040	452,200,320	243,492,480	347,846,400	2,534,400,000
					2,534,400,000	2,534,400,000	2,534,400,000	2,534,400,000	139,138,560	208,707,840	278,277,120	104,353,920	208,707,840	173,923,200	382,631,040	452,200,320	243,492,480	347,846,400	2,534,400,000
			Fog Computing	Data Aggregation	63,360	44,352	5,681,491	11,931,132	47,724,526	71,586,789	95,449,052	35,793,395	71,586,789	59,655,658	131,242,447	155,104,710	83,517,921	119,311,315	870,972,601
					2,534,400,000	1,774,080,000	1,244,246,573	870,972,601	47,724,526	71,586,789	95,449,052	35,793,395	71,586,789	59,655,658	131,242,447	155,104,710	83,517,921	119,311,315	870,972,601
	Weather	40,000	Cloud Computing	No Action	34,560	34,560	6,324,480	18,973,440	75,893,760	113,840,640	151,787,520	56,920,320	113,840,640	94,867,200	208,707,840	246,654,720	132,814,080	189,734,400	1,382,400,000
					1,382,400,000	1,382,400,000	1,382,400,000	1,382,400,000	75,893,760	113,840,640	151,787,520	56,920,320	113,840,640	94,867,200	208,707,840	246,654,720	132,814,080	189,734,400	1,382,400,000
			Fog Computing	Data Aggregation	34,560	24,192	3,098,995	6,507,890	26,031,560	39,047,340	52,063,119	19,523,670	39,047,340	32,539,450	71,586,789	84,602,569	45,555,229	65,078,899	475,075,964
					1,382,400,000	967,680,000	678,679,949	475,075,964	26,031,560	39,047,340	52,063,119	19,523,670	39,047,340	32,539,450	71,586,789	84,602,569	45,555,229	65,078,899	475,075,964
	Total number	80,000	Cloud Computing	No Action	4,723,200,000	4,723,200,000	4,723,200,000	4,723,200,000	4,723,200,000										
			Fog Computing	Data Aggregation	4,723,200,000	3,306,240,000	2,318,823,158	1,623,176,211	1,623,176,211										
	Total number	200,000	Cloud Computing	No Action	4,723,200,000	4,723,200,000	4,723,200,000	4,723,200,000	4,723,200,000										
			Fog Computing	Data Aggregation and Data Compression	4,723,200,000	727,372,800	510,141,095	357,098,766	357,098,766										
Total number		1,552,906	Cloud Computing	No Action	8,434,824,704	8,434,824,704	8,434,824,704	8,434,824,704	8,434,824,704										
			Fog Computing	Data Aggregation	8,434,824,704	2,622,834,432	1,463,333,735	862,546,313	413,501,909										
		1,328,604	Cloud Computing	No Action	8,434,824,704	8,434,824,704	8,434,824,704	8,434,824,704	8,434,824,704										
			Fog Computing	Data Aggregation and Data Compression	8,434,824,704	1,091,506,775	682,764,928	442,342,243	442,342,243										



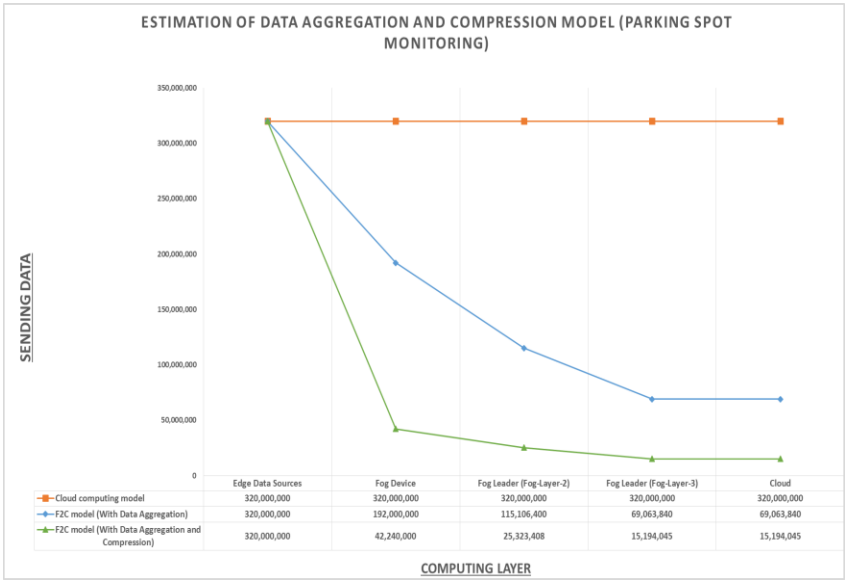
(a) Energy Monitoring category



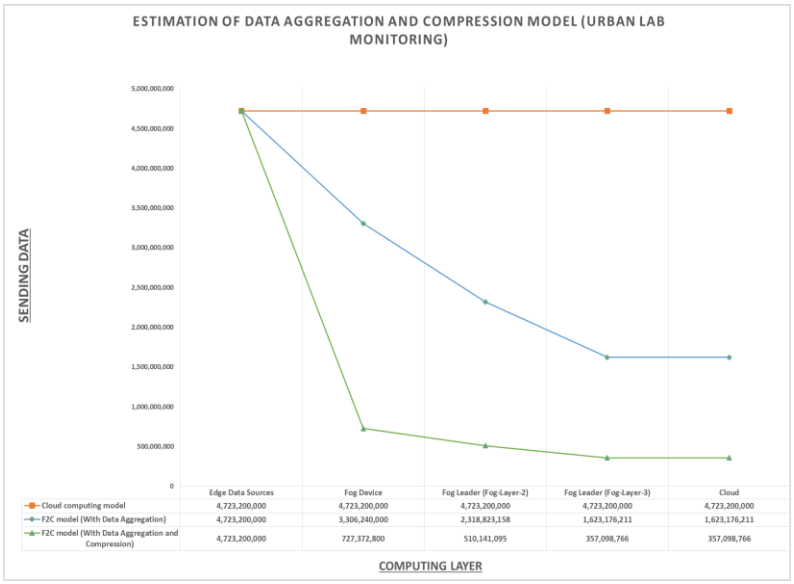
(b) Noise Monitoring category



(c) Garbage Collection category



(d) Parking Spot Monitoring category

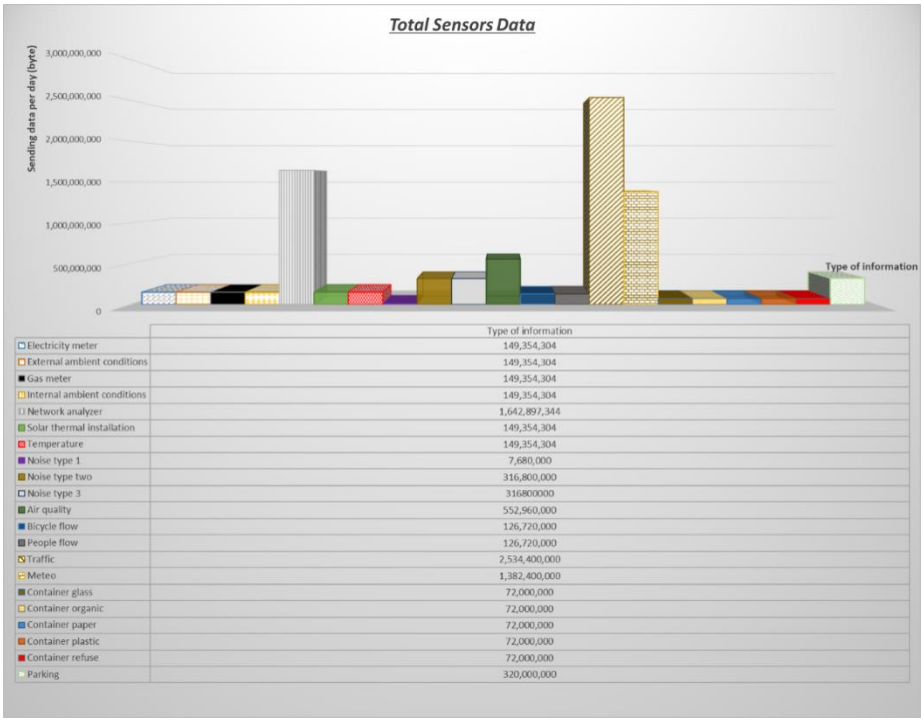


(e) Urban Lab Monitoring

Figure 5.29 Estimation of Data Aggregation and Compression Model

5.3.6Discussion of results

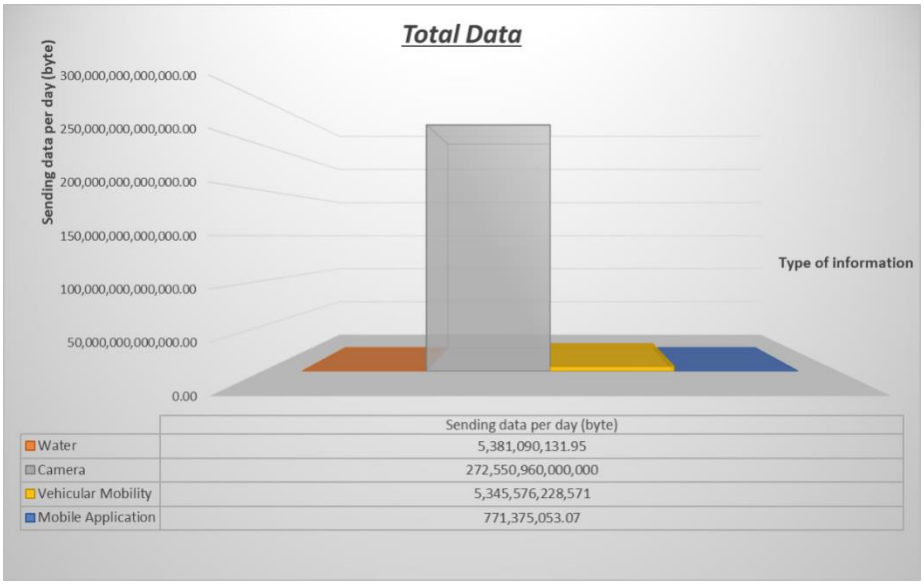
- We conclude that total amount of the data production (for each single day) will be listed in below:
  - Current sensors data by Sentilo platform: almost 0.0154 GB (15,446,712 byte).
  - Future sensors data by Sentilo platform: about 8 GB (8,583,503,168 byte).
  - Future data in Barcelona:
    - ❖ Water meter: almost 5.381GB (5,381,090,131 byte).
    - ❖ Mobile Application data: about 0.771 GB (771,106,506 byte).
    - ❖ Camera surveillance data: almost 272,550.960 GB (272,550,960,000,000 byte).
    - ❖ Vehicular mobility data: 5,345 GB (5,345,000,000,000 byte).
- We observed that traffic data type (in the Urban Lab category) has the maximum produced sensor data rate per day as shown in Figure 5.30. Oppositely, noise (type 1) sensor data type (in the Noise category) has the minimum produced data rate per day.



	Type of information
Electricity meter	149,354,304
External ambient conditions	149,354,304
Gas meter	149,354,304
Internal ambient conditions	149,354,304
Network analyzer	1,642,897,344
Solar thermal installation	149,354,304
Temperature	149,354,304
Noise type 1	7,680,000
Noise type two	316,800,000
Noise type 3	316800000
Air quality	552,960,000
Bicycle flow	126,720,000
People flow	126,720,000
Traffic	2,534,400,000
Meteo	1,382,400,000
Container glass	72,000,000
Container organic	72,000,000
Container paper	72,000,000
Container plastic	72,000,000
Container refuse	72,000,000
Parking	320,000,000

Figure 5.30Estimation of total Sensors data in Barcelona

- As shown in Figure 5.31, we realized that the camera data has the maximum data size in future Smart City of Barcelona. Additionally, mobile application data has the minimum produced data size in the future Smart City of Barcelona.



	Sending data per day (byte)
Water	5,381,090,131.95
Camera	272,550,960,000,000
Vehicular Mobility	5,345,576,228,571
Mobile Application	771,375,053.07

Figure 5.31 Estimation of total future data generation in Barcelona

- We estimated that the noise category (almost 75%) has the maximum data aggregation rate in one hand. On the other hand, urban lab category (almost 30%) has the minimum data aggregation rate.

- As shown in Figure 5.32, we appraised that total sensors data size (per day) are reduced from 8 GB to 3 GB in Fog-Device, 1 GB to Fog-Leader (sections), and 782 MB in Fog-Leader (districts) with applying the average of data aggregation rate (almost is 55%).

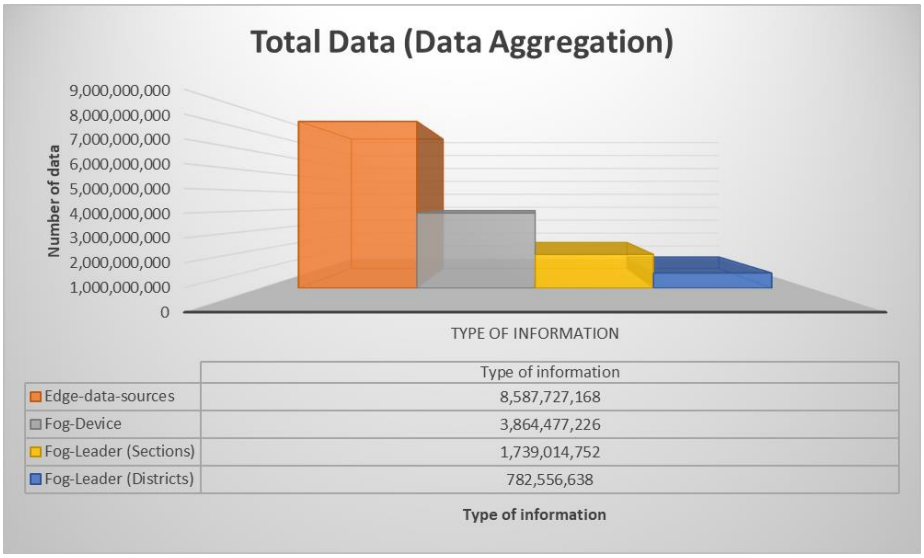


Figure 5.32 Estimation of total sensors data size with applying the data aggregation techniques

- As shown in Figure 5.33, we estimated that total sensors data size (per day) are reduced from 8 GB to 6 GB in Fog-Device, 5 GB to Fog-Leader (sections), and 4 GB in Fog-Leader (districts) with applying the data compression rate (is 22%).

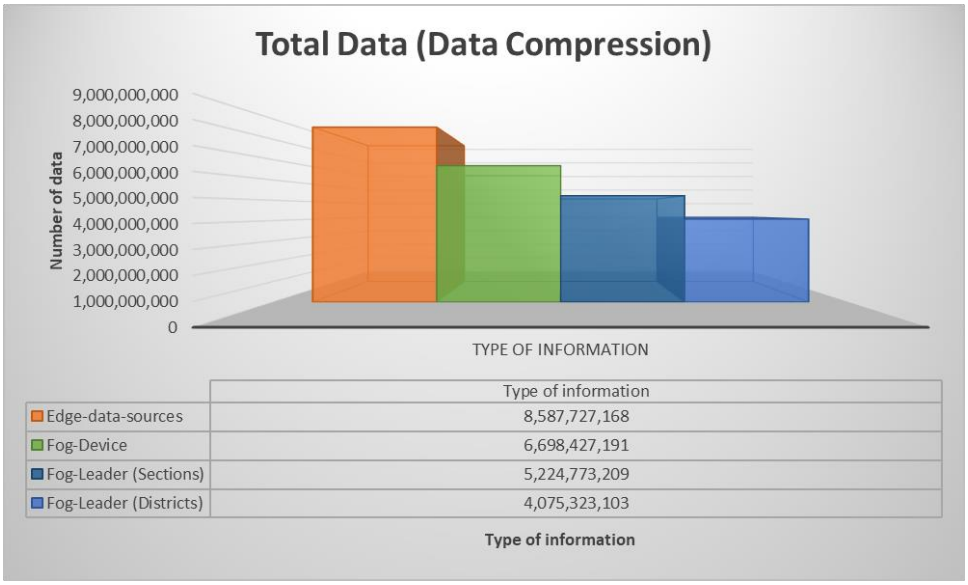


Figure 5.33Estimation of total sensors data size with applying the data compression techniques

- As shown in Figure 5.34, we find that total sensors data size (per day) are reduced from 8 GB to 1 GB in Fog-Device, 454 MB to Fog-Leader (sections), and 104MB in Fog-Leader (districts) with applying the data aggregation and compression rate (is 72%).

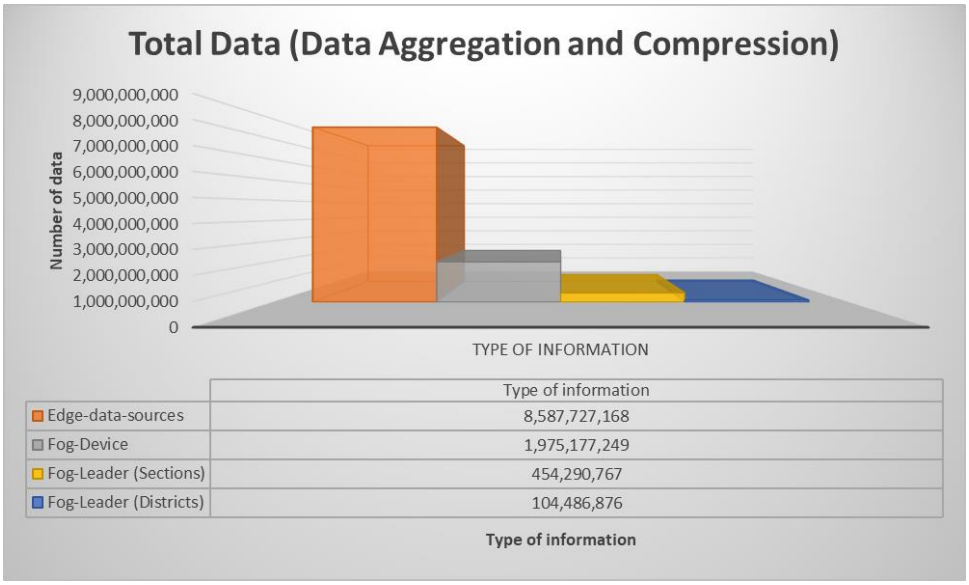


Figure 5.34Estimation of total sensors data size with applying the data aggregation and compression techniques

5.4Summary and contributions

In this Chapter, we had a deep study of all phases of the Data Acquisition block (including data collection, data filtering, data quality, and data description). We got information about the main definition, responsibility, benefits, and challenges of each phase. Finally, we conclude that there is not any complete model in the F2C scenario at Smart City to show the data acquisition block actions.

Then, we illustrated the Data Acquisition block through our F2C data management architecture. The Data Acquisition block covers with different phases which are data collection, data filtering (including data cleaning, data aggregation, and data compression), data quality, and data description. In addition, we showed data collection phase is mostly done in Fog-Layer-1 (at Edge-Data-Sources terms). However, the data collection phase is able to organize at Fog-Layer-2 (with medium performance level) and cloud layer (with minimum performance level). All other phases in the Data Acquisition block (including data filtering, data quality, and data description) can be also handled from the Fog-Layer-1 (some basic actions) and Fog-Layer-2 (more sophisticated actions), tothe cloud (advanced actions).

Next, we focused on the data collection phase. So, we estimated that the total sensors data collected numbers (including current and future sensors data through Sentilo platform) in the city of Barcelona. So, we concluded the Sentilo platform (in Barcelona) generated almost 0.0154 GB (15,446,712 byte) sensors data in the current situation and 8 GB (8,583,503,168) per day sensors data in the future situation (with full coverage of sensors). In addition, we extended the type of information for the future Smart City of Barcelona. We estimated that 272562.457 GB data per

day (including 5.381GB of water data, 0.771 GB for mobile applications data, 272,550.960 GB for camera surveillance data, and 5,345 GB for vehicular mobility data) will be added to the future data collection in Barcelona city.

Then, we proposed some filtering techniques (data aggregation and data compression) through the data filtering phase in our F2C data management architecture. So, in the case of Barcelona Smart City we have shown by applying redundant data elimination that, in some cases, the data reduction rate reaches 75%. Additionally, by applying data compression, the data reduction rate increases to up to 78%. Finally, we have explored that the total efficiency rate, by applying both redundant data elimination and data compression, moves to almost 92%, in some cases. Although many other data aggregation techniques could be easily applied in this architecture, these two basic techniques are enough to illustrate the facility and effectiveness of such optimizations in our model.

We have listed the main contributions of this Chapter as shown in below:

- To make the survey about the data acquisition block and related phases (including definition, related works, challenges, and responsibilities) in any scenario.
- To illustrate data acquisition block and their phases through F2C data management architecture [206-208].
- To calculate the current sensors data production through Sentilo platform in the Smart City of Barcelona[209].
- To estimate the future sensors data production through Sentilo platform in the Smart City of Barcelona[209].
- To estimate the future data production (with more category of information) in the Smart City of Barcelona.
- We have shown the facility and effectiveness of applying some data filtering techniques (including data aggregation and compression) through the collected data in the Barcelona Smart City [206, 207].
- To show the efficiency rate after applying the redundant data elimination and data compression techniques in the Smart City of Barcelona [206, 207].

The data acquisition block (through our F2C data management architecture) can provide some desirable advantages as described in below:

- Organizing and managing the data collection phase in the Smart City.
- Providing some data filtering facility (such as redundant data elimination techniques) for the collected data in the Smart City.
- Appraising the data quality for the collected data in the Smart City.

- Describing the collected data (including the production time, ownership, and etc.) for the future use in the Smart City.
- Providing compressing techniques to compact the size of the collected data over the network communications in the hierarchal distributed environment at Smart City.
- Enabling fast real-time access.
- Providing traffic reduction at different level of distributed hierarchal architecture.
- Having a better level of data quality by applying data filtering techniques.

We proposed some publications for this Chapter in the reputable venues which can be sorted in the below:

- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, "Estimating Smart City Sensors Data generation and Future Data in the City of Barcelona", in IFIP Med-Hoc-Net 2016, Vilanova i la Geltrú, Barcelona, Spain, June 2016.
- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, X.Yin, C.Wang, "A Data LifeCycle Model for Smart Cities", IEEE conference on ICTC 2016, Korea, October 2016.
- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera,"A Novel Architecture for Efficient Fog to Cloud Data Management in Smart Cities", IEEE ICDCS 2017, Atlanta, USA, June 2017.
- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera,"Fog to Cloud Data Management in Smart Cities", IEEE FTC 2017, Vancouver, Canada, November 2017.

Chapter 6:

The Data Preservation Block

Digital information is generated constantly with an abundant of big data scenarios (such as smart cities) every moment around the world. This vast number of generated data must be recorded in data repositories for any future usage. Therefore, we can conclude that data preservation is a key in today´s information century [210]. In addition, cloud computing generally provides a variety and an unlimited amount of resources (through their data centers) to handle all the preservation requirements in any scenario (such as smart cities).

In our view, the Data Preservation block is the responsible for data storage and the performance of any eventual action related to data curation or data classification. This data is ready for future dissemination, or for further processing.

In this Chapter, we aim to present the following concepts: In section 6.1, we described the phases in the Data Preservation block. And, we say about all definition, state of the art, and challenges of each phase. In Section 6.2, we go beyond to the Data Preservation block in F2C. Then, we introduce the proposal of all phases in the Data Preservation block (including data classification, data storage, and data description). In section 6.3, we illustrate the data preservation block in the Barcelona Smart City. In Section 6.4 section, we estimated the number of the stored data in each data storage levels (from Fog area and sections to districts). Finally, we conclude the main contributions of this Chapter.

6.1 Phases in the Data Preservation Block

Digital information is generated constantly with abundant big data scenarios (like smart cities) around the worldwide in every moment. This huge number of generated data must be recorded in data repositories for any future usage. Therefore, we can conclude that data preservation is a key in this information century[210]. In addition, normally cloud computing provides varieties and unlimited resources (through their data centers) to handle all preservation requirements in any scenario (like smart cities).

In our view, the Data Preservation block is the responsible for data storage and performing any eventual action related to data curation or data classification. This data is ready for future publication or dissemination, or for further processing.

Figure 6.1 depicts that Data Preservation block comes with three main phases; Data Classification, Data Archive, and Data Dissemination. Related phases will be explained in details in the following sections.

Figure 6.1 Phases in the Data Preservation Block

Main objectives of this block to be discussed are the following:

- To classify and sort all received data before storing time.
- To store data for the permanent and temporary reasons.
- To publish data for further usage by end-users and provide an effective interface to facilitate data access.

6.1.1 The Data Classification phase

The Data Classification phase is the first phase of the Data Preservation block (as shown in Figure 6.2). This will be discussed in more details in the following subsections.

Figure 6.2 The Data Classification phase in the Data Preservation Block

6.1.1.1 Definition

Data Classification provides an efficient organization for storing data. Generally, data classification follows by a specific standard (including a manual or systematic standard). Mainly this standard provides some facility to indicate data properties, data valuation and etc. There are existing work related to this, where the author has attempted to organize data in regards to their value (data valuation) to be stored in exact performance level storage devices [211]. In this work, the author created a method which links the business and storage infrastructure together. Therefore, this link can classify stored data by determining of their value.

6.1.1.2 State of the art

There is a different view to classify data for storing time. First, there is some related work which the author tries to organize data regarding their value (data valuation) to be stored in exact performance level storage devices [211]. In this work, the author made such a kind of a bridge method to create link between business and storage infrastructure. Therefore, this bridge can classify data stored data by determination of their value. The author has also mentioned that data valuation can be categorized into two main categories: i) File level: The file system represents the data which is processes granularity in the business server; ii) A large storage: block-based solution handles all the operations (including backup, restore, and etc.) instead of file based in datacenter. This paper depicts several related techniques to handle data classification through file level-based

or block-based solutions. Second, in another view, as we discussed in the previous chapter, there is a specific type of metadata which is able to attach more information about the published data (including the name of the data set, the unique identity, publisher, publisher organization, publishing time, and the link for accessing the data). Therefore, in [190] gave a particular example to show that this metadata provides facility to connect to the data by the following link: i) direct access to data (e.g. local open data platform); ii) accessing data after following user authorization (e.g. quasi-sensitive data and private cloud store); iii) indirect access to data (e.g. connecting to another open data platforms in other Smart City through your open data platform in your Smart City). Indeed, the author (with focus on cloud computing environment) [212] mentioned that a variable data classification index has been set to classify stored data in cloud servers. The proposed index is able to determine the value of this index dynamically according to the specifications of stored data. In addition, the index utilizes three main parameters which are data confidentiality, data integrity, and data availability.

In our view, some additional descriptive information could be also attached to this data related to the archiving policies, such as access permissions, privacy, expiry time, or sharing, use and reuse capabilities. In this phase, data provenance or data versioning could be considered.

6.1.1.3 Objectives and challenges of an effective Data Classification phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data classification phase as part of a data management architecture. Some of these challenges, such as those associated with the automatic data valuation (as a concept of organizing data classification automatically), are out of the scope of this research thesis. The main objectives and challenges are the following:

- Making an efficient classification data methods for indexing and searching the data.
- Design an appropriate criteria (as following manual or systematic standard) for data classification (including data valuation, data properties and etc.)
- Classifying and organizing data before storing, according to the city's requirements and the business model.
- Classifying the stored data (including new and old data) with some specific criteria (such as data production time) to ease of keeping track of their location according to the age (just collected, least recent, and historical)
- Adding some additional metadata regarding storage, such as expiry time, usage and reuse capabilities, security level, and so on.
- Implementing the corresponding management techniques in order to implement any data versioning, data lineage or data provenance.

- Data classification can get helps to increase the efficiency of data protection procedure and decrease energy consumption with eliminating any further unnecessary process in future.
- To aim to organize and prepare data for efficient storage, by applying some optimization, such as classification, arrangement, compression, etc.
- To classify data with specific standard which is mostly related to the value of data.
- Create and manage the preservation metadata to describe process and preservation actions for digital objects.
- These objectives contribute in simplifying the data challenges of :
    - Data Volume, because huge numbers of the data stored can be organized for any future use by applying the data classification techniques.
    - Data Variety, because the huge different type of the stored day can be categorized for any future use in the storages by demonstrating the data classification techniques.
    - Data Velocity, defining the specific criteria (in the data classification) can be quickly helped to organize the high-speed rate of the data creation at storing data over time.
    - Data Variability, providing the specific policies (in the data classification) can be efficiently helped to manage the different version of the produced data at storing data over time.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. We just show how easy and efficient is applying anF2C object store service model to store data in the distributed environments. This will be described and discussed in subsection 6.2.

6.1.2The Data Archive phase

The Data Archive phase is coming after Data Classification phase as shown in Figure 6.3. We will present more details about this phase as the following subsections.

Figure 6.3 The Data Archive phase in the Data Preservation Block

6.1.2.1 Definition

The digital preservation (archiving) means to save digital data for future access and usage regarding the user demands. There is some important parameters (such as robustness, safe, secure, and etc.) to have better performance for the archiving digital information. In addition, life time parameter plays an important role in the digital preservation. Currently, there is some view about distributed digital preservation which it goes beyond how digital data must be saved in the distributed storages over the network in any scenario.

6.1.2.2 State of the art

In[210], the author mentioned that digital preservation can be categorized in three different categories as shown as below :

- Short-term preservation: This type of information is valid and accessible with a defined period of time (because of changes in technology) in fact.
- Medium-term preservation: This type of information is valid and accessible beyond changes in technologies but it does not consider for the indefinite access.
- Long-term preservation: This type of information is valid and accessible without any specific limitation.

Plus, the same author ([210]) believes that digital preservation must provide: i) reliable devices for stored data; ii) Make a backup copy for data corruption; iii) Preserving Metadata. In particular, preserving metadata plays an important role for data archive. In addition, the metadata archives present the archived features resources for future usage.

In [213], author proposed that file system prepare some specific procedure for data archive which is able to store, retrieve, and update data. So, this procedure can be handled the available space on the device(s) which contain it. In addition, distributed file system (DFS) is kind of centralized view to handle file system beyond storing data infrastructure. In addition, the different type of distributed file system are Lustre, Kosmos, Hadoop, Google file system and etc. And then, other author mentioned that different type of file system can be seen in cloud computing environment which is distributed and parallel databases, cloud data serving systems, and data freshness[214].

Recently, an author proposed an object store service for a Fog/Edge computing infrastructure based on IPFS and Scale-out NAS[191]. The author used two different protocol to depict an object store service in Fog/Edge computing. The first proposed model is designed under IPFS (InterPlanetary File System) storage system. This model showed how data will be stored in the Edge of the network. Plus, this model used two famous protocols for scaling a large number of nodes (including Kademlia DHT and BitTorrent protocol). The DHT is a place for metadata management and object location access in IPFS at the edge of the network and is allocated in Fog computing node (which is on top of Edge nodes). Regarding the author's argument, this model has

some challenges to read an object stored data locally through accessing the DHT in terms of accessing time and reading data from local data stored. Therefore, the second model is proposed to coupling IPFS and Scale-Out NAS systems. This model is kind of extension of a previous model for a large number of nodes. The idea provides the facility for each IPFS node of one site to access data stored by others nodes at the same time as shown in Figure 6.4.

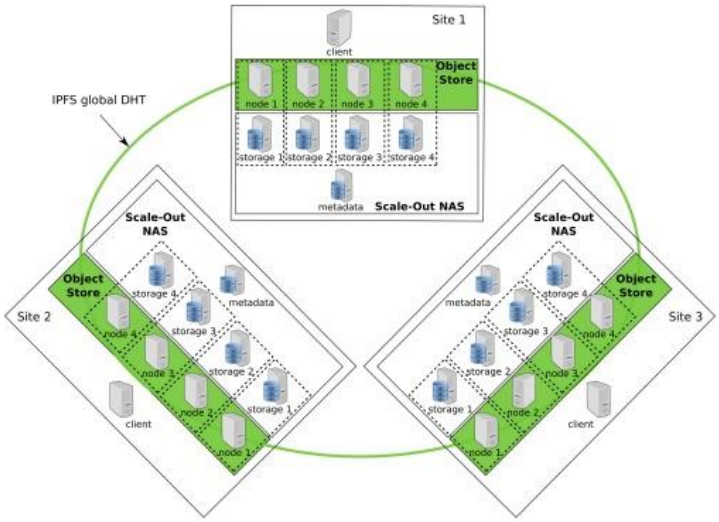


Figure 6.4 Coupling IPFS and Scale-Out NAS systems[191]

6.1.2.3 Objectives and challenges of an effective Data Archive phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data archive phase as part of a data management architecture. The main objectives and challenges are the following:

- Providing the reliable, secure and accessible platform to store all produced data for long term preservation objectives in the city.
- Storing (large sets of) data collected and processed in the city. Data will be stored in temporal sites, distributed along the city, and a selection of data (aggregated) will be permanently stored in the cloud.
- Organizing files and directories, and keeping track of which areas of the media belong to which file and which are not being used as the file system.
- Providing efficient access to data depending on their age (including real-time, least recent, and historical data) for short-, medium- and long-term archiving of large volumes of digital resources in the digital repositories.
- The repository architecture enables replacement and continuous upgrades of individual system components (subsystems, filesystems etc.) following the emergence of new technologies.

- This phase is responsible for the long term preservation, but also responsible for some additional tasks, such as data cleaning according to the corresponding expiry time, or implementing other business related policies.
- These objectives contribute in simplifying the data challenges of :
    - Data Volume: Huge numbers of data are generated in the city. There are many challenges and difficulties to save efficiently the produced data (in terms of real-time, least recent, and historical data) in the distributed environments.
    - Data Velocity: Data is growing fast. So, the frequency of update, accessing data (such as real-time), and scalability are challenging concepts through distributed environments.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. We just depict how fast and useful is to store data through an F2C object store service model in the distributed environments. This will be described and discussed in subsection 6.2.

6.1.3The Data Dissemination phase

The Data Dissemination phase is the last phase in the Data Preservation Block. This phase will be explained in more details in the next subsections.

Figure 6.5 The Data Dissemination phase in the Data Preservation Block

6.1.3.1 Definition

This phase is able to provide the facility for end-users to look forward to the produced data regarding their demands. In general view, the file system creates an Application Programming Interface (API) to get helps to end-users to save and access data over distributed storage devices and provide storage location transparency as well. Recently, distributed file system (DFS) build a specific mechanism (including the logical view of directories and files) to look for required data without any information about physical resides of data in the network [215]. In addition, distributed file systems are reported by several advantages for data disseminations which are listed as below:

- Centralized view but distributed over networks.
- Easing of update and open any file on any machine on the network.

- Ability to backups and centralized management.
- Some additional facilities such as user mobility, location transparency, location independence and etc.

6.1.3.2 State of the art

In [216], author mentioned that distributed file system can be demonstrated with Stand-alone DFS Name Space (\\ServerName\RootName) and Domain-based Namespace (\\DomainName\RootName). The point is that the first model includes the name of the server name in the path but the last one starts with the domain name in the path. In addition, in Smart City environment, the author mentioned that CKAN is a great solution for disseminating data in open data platform [190]. In addition, In [217], they mentioned that the naming strategy is presented in distributed environments. This strategy has three main components (including certification, hashing, and identification) which provide facility to manage resources in the distributed environments. The certification is responsible to register any connected device to the distributed architecture. The Hash table is able to follow the naming scheme protocol for the distributed environment to store the name of devices into the table. Indeed, the identification gets help to the hash table to find the hash value of devices in the distributed environment. Finally, [191] said that an object store service model provides facility to manage the data location in an object store. So, the management of data location proposed by "write" and "read" scheme.

In our view, this phase is the natural interface with the end-user for stored data. Additionally, this data can also be considered for processing, as part of the Data Processing block.

6.1.3.3 Objectives and challenges of an effective Data Dissemination phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data dissemination phase as part of a data management architecture. The main objectives and challenges are the following:

- Creating an efficient and flexible interface to get help to the users for accessing to the stored data in the distributed environments.
- Data is distributed across the network. So, managing data (across the distributed network) provides some splendid facility for keeping track of their data location according to their data production time.
- Providing and managing the access permissions, privacy, expiry time, or sharing, use and reuse capabilities
- Designing a set of policies for the data expiry and life time in the distributed system. / Having a specific sharing policy for the distributed system

- These objectives contribute in simplifying the data challenges of :
    - Data Volume, because huge numbers of the data stored can be shared for any future use.
    - Data Variety, because the huge different type of the stored day can be accessible to the users.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. We will explain how easy it is to find the data in the distribute environments through an F2C object store service model to store data. This will be described and discussed in subsection 6.2.

6.2The Data Preservation Block in F2CSmart City

As shown in the previous Chapter, there are a huge number of data production over times in smart cities. In addition, F2C makes facility to provide real-time, least-recent, and historical data for data stakeholders. So, the Data Preservation block (through the F2C data management architecture) must be able to manage all complexity of the high data production (including real-time, least-recent, and historical data). All above information highlighted that it is necessary to organize the complex system (including addressing data and sources, writing data, searching data, and etc.) of the Data Preservation block in F2C with an efficient interface. This interface must be able to get access to different data types in terms of time dimension (including real-time, last-recent, and historical data). Plus, the interface must have possibility to find the appropriate data for the end-users.

The F2C data management architecture can handle all phases of the Data Preservation block from Fog to cloud scenario. As shown in Figure 6.6, the Fog-Layer-1 covers with low capacity of resources to manage all phases in the Data Preservation block (including data classification, data archive, and data dissemination). And then, Fog-Layer-2 has more resources to organize all phases in the data preservation block. Indeed, cloud prepares the highest level of data sources (with almost unlimited resources) for all the related phases in the Data Preservation block.

As shown in Figure 6.6, the light color of all phases are presented that the data actions (including data classification, data archive, and data dissemination) in the minimum progress at that layer. Oppositely, the dark color of all phases are showed that the data actions (including data classification, data archive, and data dissemination) in the maximum progress at that layer.

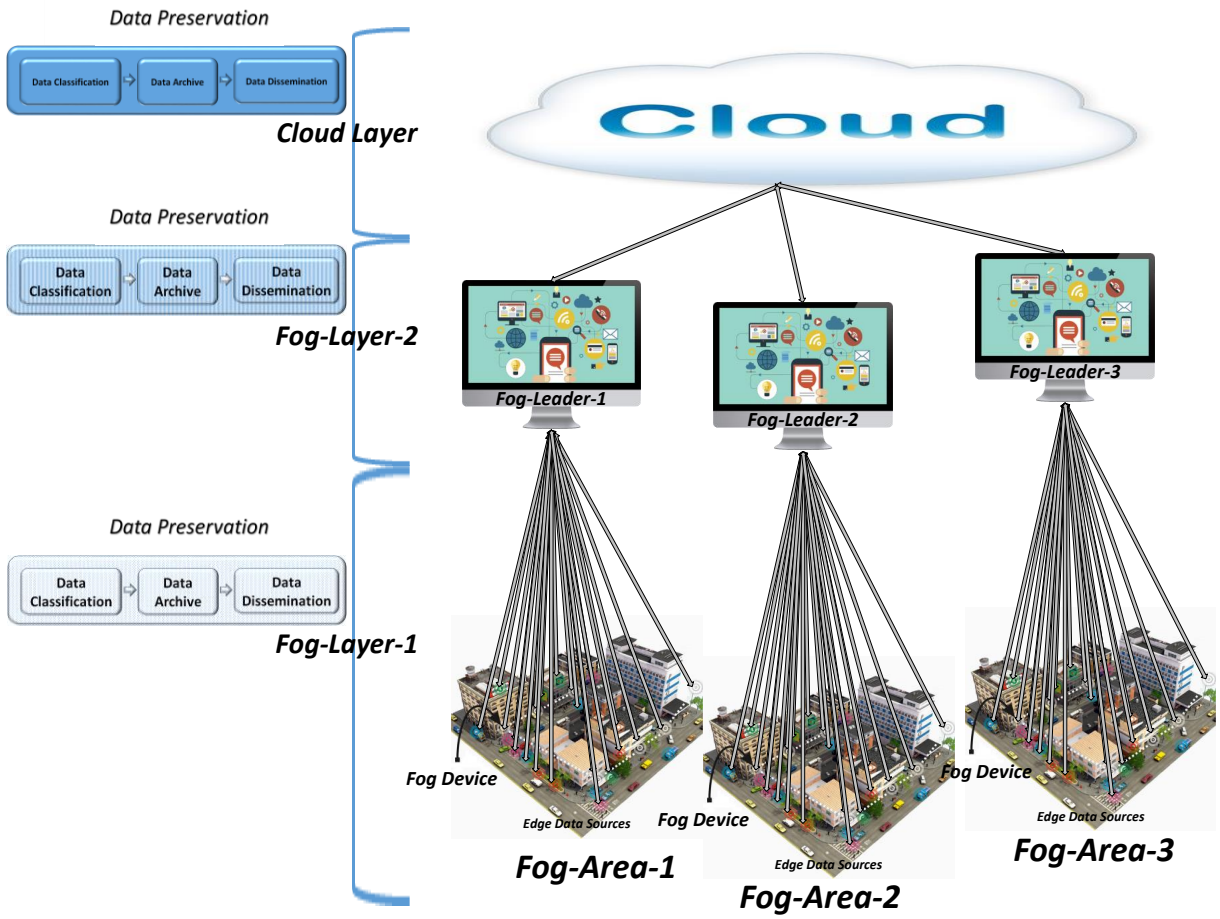


Figure 6.6 Description Scenario of Data Preservation Block

In general terms, the data preservation block in the F2C architecture is the responsible of two main tasks:

- Managing the storage of the collected data through the different hierarchy layer, and
- Providing an efficient interface for the data usage, by maintaining the appropriate directory structures, and providing a flexible access interface according to the smart city applications requirements. This is named the F2C object Store Service model

These two main tasks are described in the following subsections.

6.2.1The F2C data storage architecture

In Figure 6.7, we illustrate that there are different level of the data storage in F2C as shown in below:

- Data storages in Fog-Devices (at Fog-Layer-1);
- Data storages in Fog-Leader (at Fog-Layer-2);
- Data storages in data centers (at cloud layer).

In fact, each layer has different capacity to store data in F2C. So, the data storage capacities are in the minimum level at Fog-Devices. And, the capacity of the data storages are getting more in the Fog-Leader at Fog-Layer-2. Finally, the capacity of the data storages are almost in the unlimited size in the data centers at cloud layer.

The F2C data management architecture make further facility to organize the frequency of update for the stored data (from fog layers to cloud). It means that each produce data will able to move to the upper layer after some specific time slots. The time slots can follow the business requirements and city managers policies.

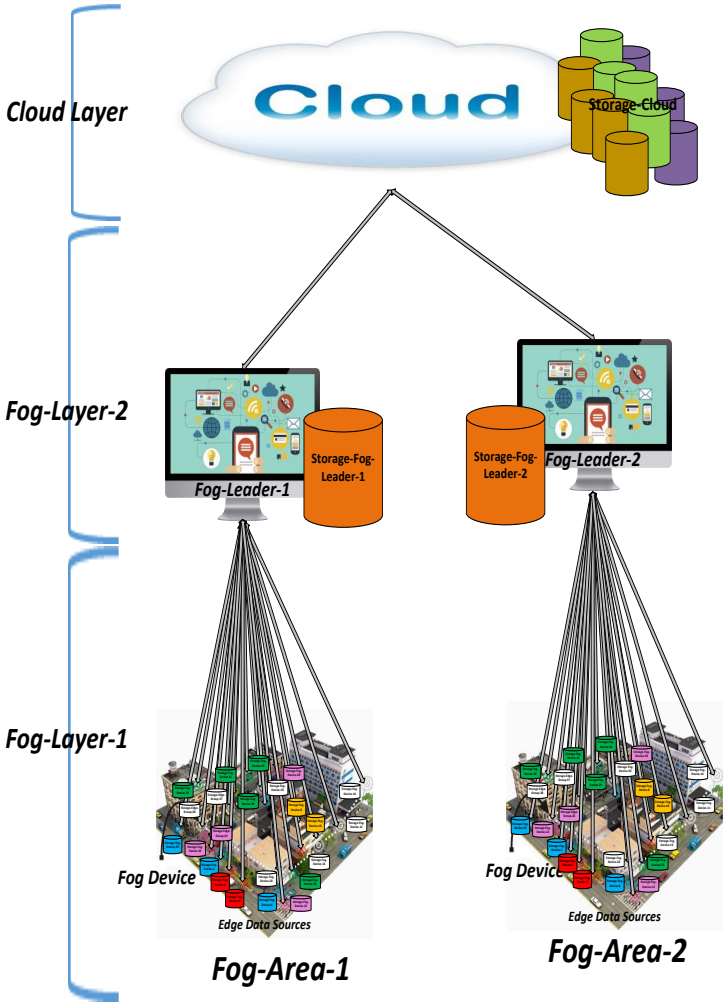


Figure 6.7 Data Storage level in the F2C data management architecture

The real-time data generate in Fog-Layer-1 which are closest data to the physical devices and users in the smart cities. Plus, the real-time data is able to move to the upper layer (in Fog-Layer-2) after some defined time by the city managers. The available data (in Fog-Layer-2) is namely called least-recent data. Similarly, the least-recent data (at Fog-Layer-2) will pass to highest level (cloud layer) after some defined time by the city managers. The cloud data consider as historical data.

F2C data management architecture provides facility to get access to different level of the stored data (including real-time, least-recent, and historical data) from fog to cloud environments. If a specific (or critical) data is required at real-time from a close location, it is obtained from the source (distributed); however if more complete data set is required, probably least recent, it is obtained from upper levels (thus with higher capacities). In addition, if the complete history of data is requested, probably historical data, it is available in more centralized nodes at top layer. Therefore, we realize that different type of the stored data can be reachable in the different layers of F2C as shown in below:

- Fog-Layer-1:
  - Fog-Device: As shown in Figure 6.8, the Fog-Devices is responsible to real-time data.
- Fog-Layer-2:
  - Fog-Leader: As shown in Figure 6.8, the Fog-Leader has the last-recent data.
- Cloud:
  - Cloud storages: As shown in Figure 6.8, the cloud storages provide the historical data.

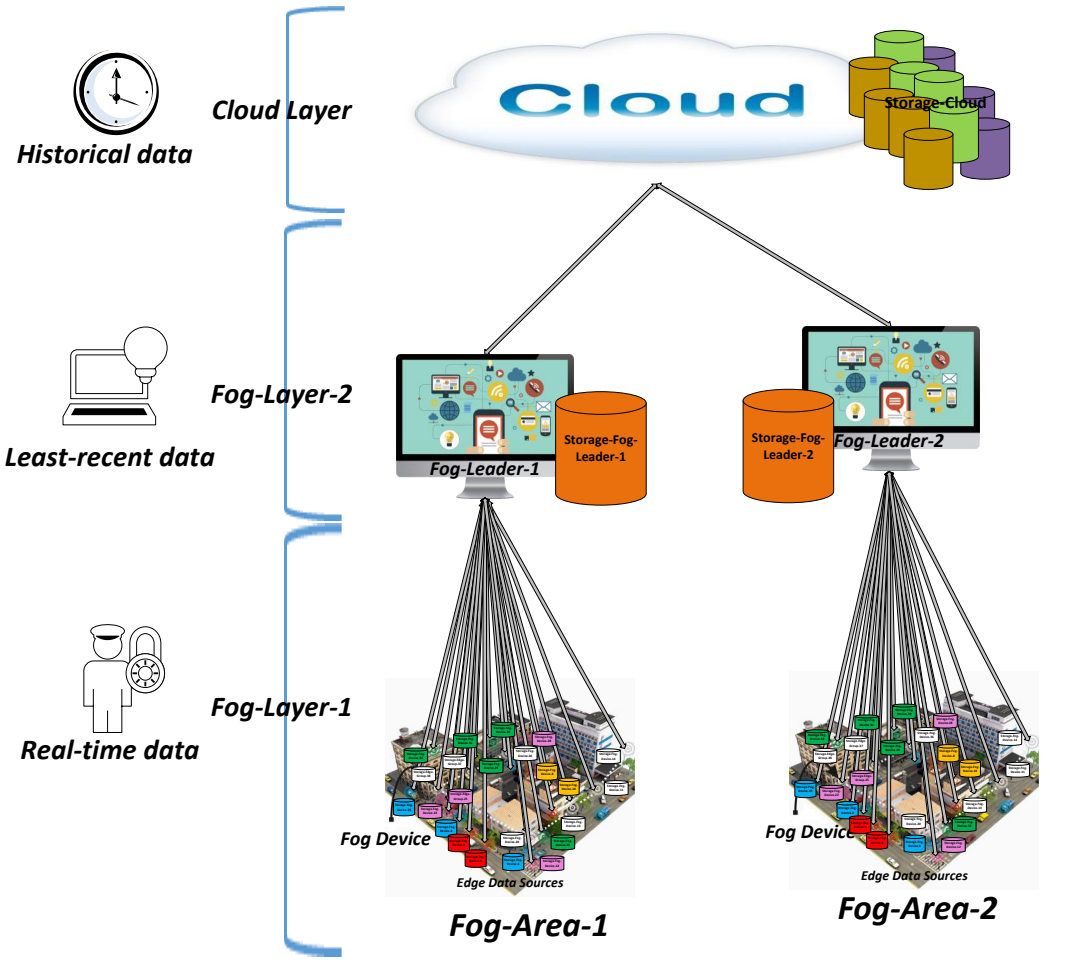


Figure 6.8Different types of the stored data in the F2C data management architecture

6.2.2 TheF2C object Store Service model

The F2C data management architecture follows Content Delivery Network (CDN) to put name for the stored data in the distributed environment. In the CDN naming, the data will be stored and called by their name. So, we define three dimension for the stored data which are location, time, and type of information. Indeed, we are able to call and search the request data with this three parameters in distributed hierarchal storages (including Fog-Device, Fog-Leader, and cloud environments).

The object store service model aims to manage the data location in an object store. So, the management of data location goes beyond two main concepts which are "write" and "read". In one hand, write concepts present part of an object store service model to depict how data will be stored in F2C model which we say more details in this section. On the other hand, read concepts introduce (as another part of an object store service) how data will be existed in our hierarchal distributed model through F2C computing which we give more details about this part in the last chapter.

In this section, we describe our scenario for the hierarchal distributed object store service model through F2C computing. And then, we have focus about write part at Fog-Layer-1, Fog-Layer-2, and cloud computing as show in below subsection.

6.2.2.1 Description scenario

As we mentioned before, DHT makes facility to save address of the information of connected devices in the related hash tables. So, our scenario has two main objects which is defined as shown as below:

- Client: The client is someone or device which requests to save or read information in a specific Edge-Data-Sources layer;
- DHT (Distributed Hash Table): DHT is responsible to metadata management and Hash table of our scenario. So, in case the data is not available in the requested node, the DHT goes to other related hierarchal distributed DHT to find the address of the data. Indeed, if we do not find the data, we send a message (like “not found”) to the user.

In our scenario as shown in Figure 6.9, DHT is located in all Fog-Devices (Fog-Layer-1), Fog–Leaders (in Fog-Layer-2), and cloud layer. The idea is that DHT (in cloud) will be updated by all below DHT in our hierarchal distributed model (including Fog-Devices, and Fog-Leaders) frequently. The frequent update schedule is depend on the business requirement and city managers policies.

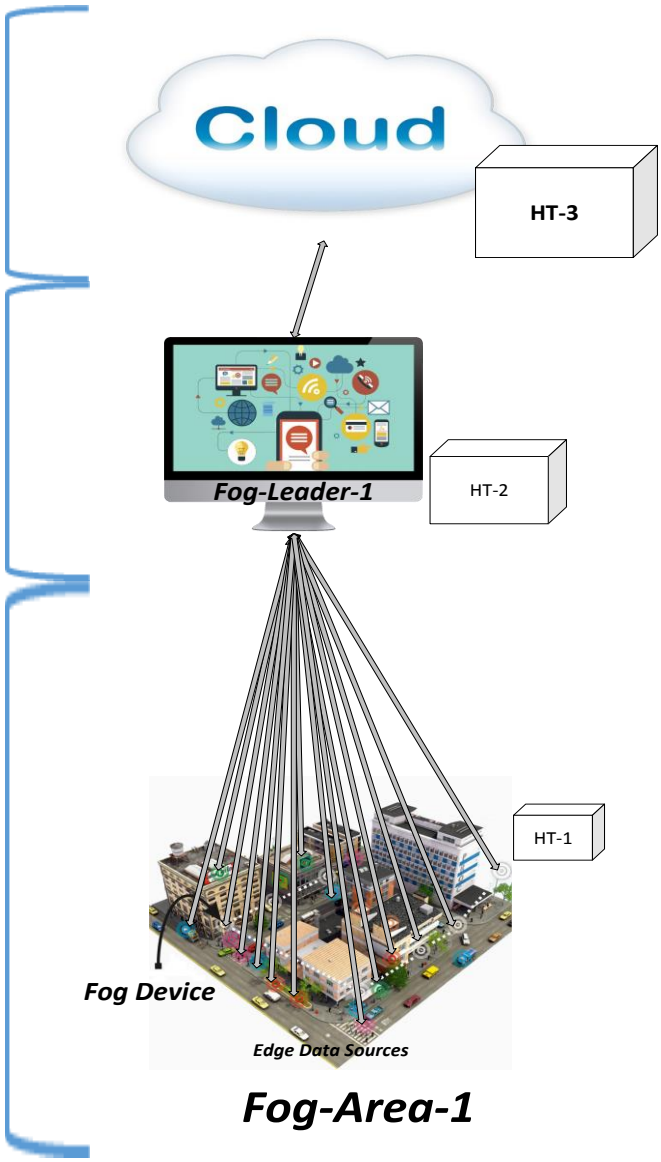


Figure 6.9 The object Store Service model for F2C computing model

As we mentioned on top, DHT is responsible to all metadata and Hash tables of our data in the F2C model. So, in this case we have different cases as shown as below:

- Current device (like a specific sensor) willing to store some information in the distributed environment;
- A new device (like a new sensor) aims to join to the closest Fog-Device to write their produced data in F2C architecture;
- A new Fog-Device (including with the specific device) tries to send the generated data in the F2C architecture.

All above different situation will be described in the following subsections as shown in below:

6.2.2.2 Write by a current sensor to the Fog-Device

The first case is that there is a current sensor in our network which is already registered in our F2C data management architecture (in particular in DHT). And, the sensor generates some data to store in the nearest hierarchal distributed storage (in particular nearest Fog-Device) through the F2C data management architecture. Therefore, as shown in Figure 6.10, data will be sent to the nearest Fog-Device layer to store in the particular storage at this Fog-Device. So, the generated data will be stored in the Fog-Device and then the metadata information will be saved in the DHT at Fog-Layer-1 for future usage. Finally, all upper hash table will be update frequently with this new table of the DHT at Fog-Layer-1. Similarly, the stored data will be sent to upper storage layers (including Fog-Layer-2, and cloud) regarding the frequency of our update schedule.

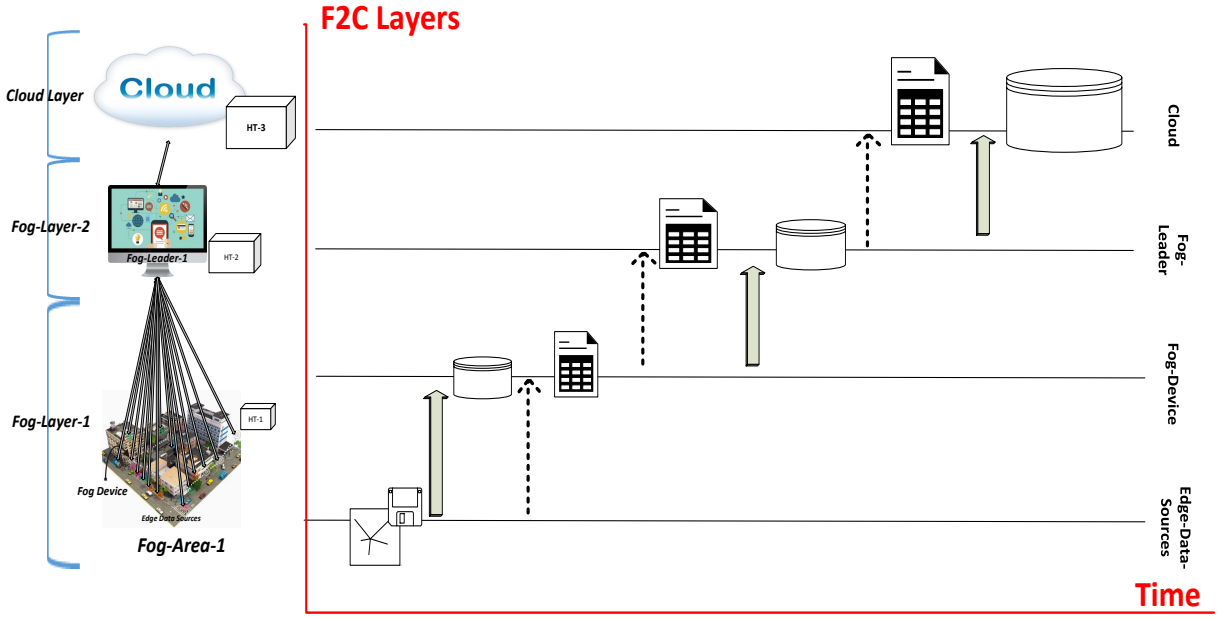


Figure 6.10 Writing schema by a current sensor to the Fog-Device

6.2.2.3 Write by a new sensor to current Fog device

The next case is that a new sensor (as a physical device) wants to register to the F2C data management architecture as shown in Figure 6.11. In this case, first the sensor will be joined to the closest Fog-Device. And, the sensor will be registered their device information into the DHT of Fog-Device. Additionally, all upper DHT will be updated with this new information after some particular time. After all this procedure, the new sensor is able to save the produced data in the F2C data management architecture (like the previous procedure as shown in Figure 6.10).

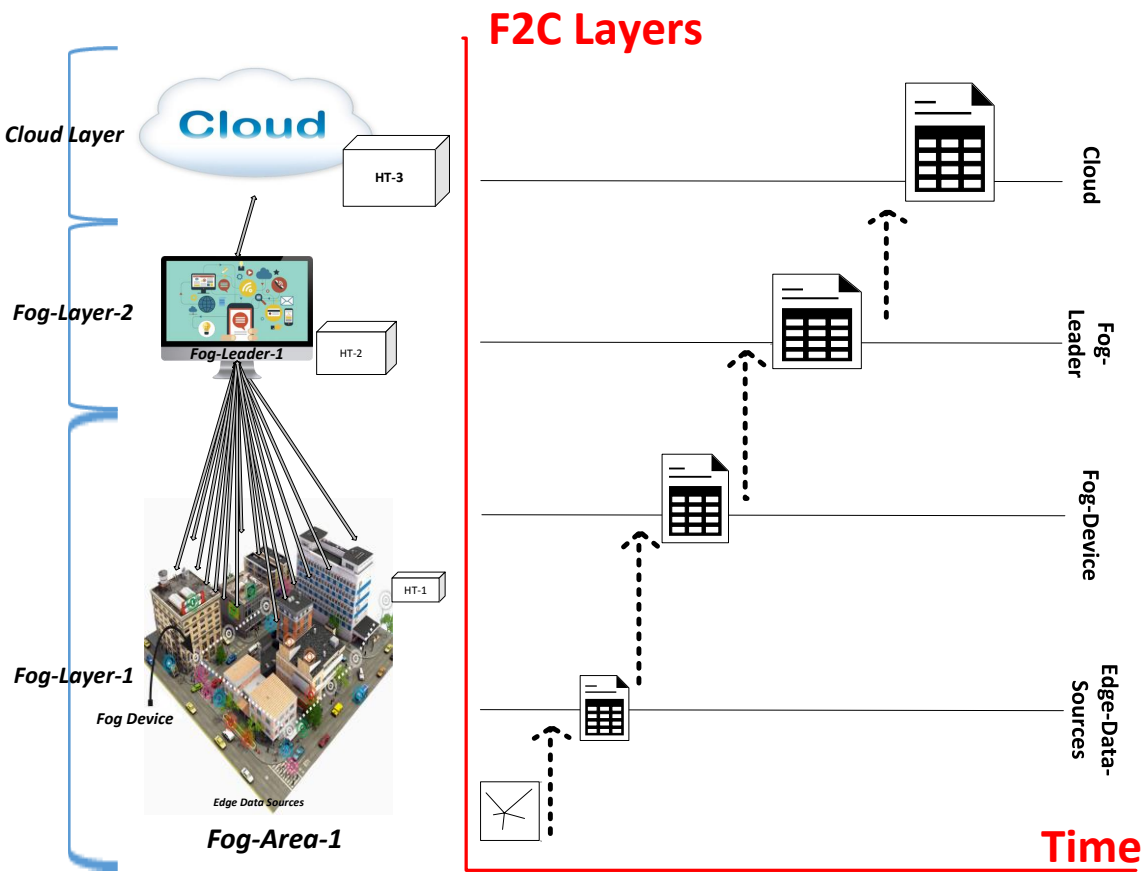


Figure 6.11 Writing schema by a new sensor to current Fog device

6.2.2.4 Write by a new Fog device to Fog-Layer-1

Similarly, as shown in Figure 6.12, if a new Fog-Device requests to join to the F2C data management architecture, first the Fog-Device must be sent the their hash table information to the closest hash table in Fog-Layer-2. Additionally, the DHT (in cloud layer) will be updated by DHT in Fog-Layer-2 as a specific update time. Indeed, if any data stored in this Fog-Device, the stored data can be transferred to the storages of the Fog-Layer-2 layer (and then the cloud) regarding the update policy of the F2C data management architecture.

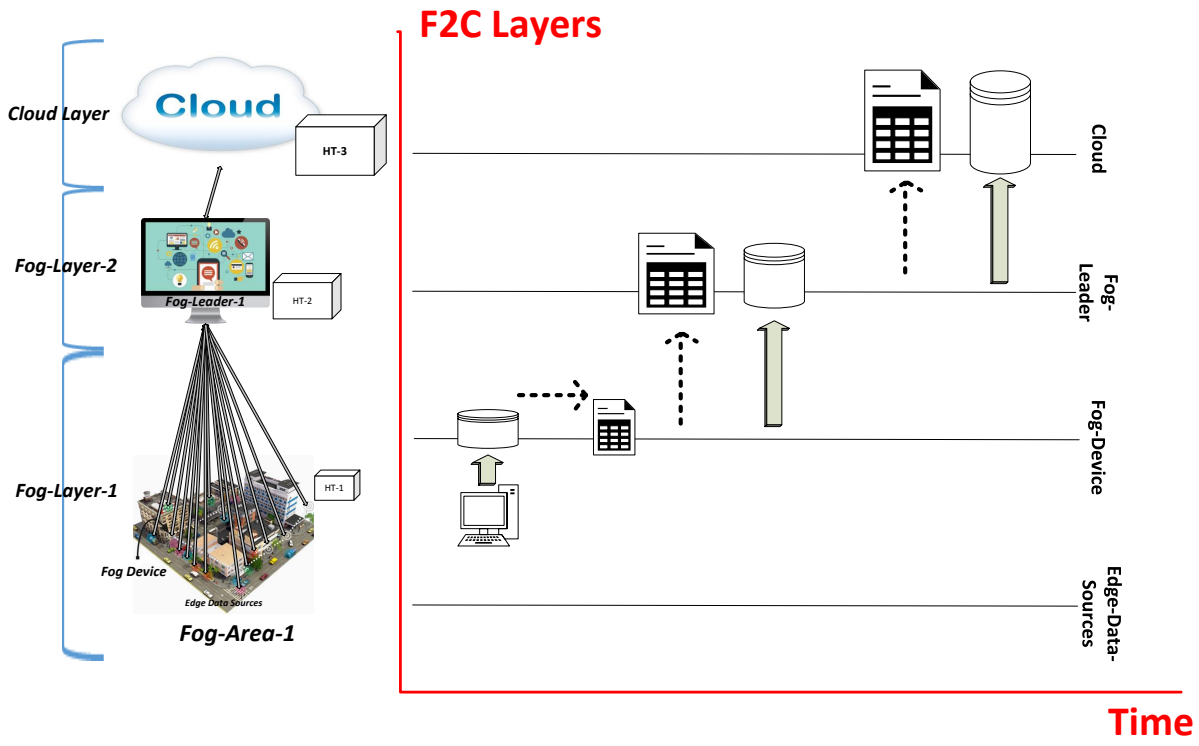


Figure 6.12 Writing schema by a new Fog device to Fog-Layer-1

6.3Experimental results: Estimating data preservation in Barcelona

In this section, first we describe our scenario in real Barcelona Smart City. Then, we calculate the data storage size in F2C (including the minimum, maximum, and total data storage capacity in F2C). And then, we argue about the discussion of the result.

6.3.1Scenario Description

In particular data archive phase in Barcelona, we depicted that there is different level of storage at Fog-Device, Fog-Leader for the sections of Barcelona (in Fog-Layer-2), Fog-Leader for the districts of Barcelona (in Fog-Layer-3), and cloud as shown in Figure 6.13. In addition, each layer transfers archived data from down to up at the specific defined time (as frequency of update). So, the different layers of data storage can be listed in below:

- Fog-Layer-1:
  - Fog-Device: is responsible to real-time data.
- Fog-Layer-2:
  - Fog-Leader: has last-recent data (related to the sections of Barcelona).

- Fog-Layer-3:
  - Fog-Leader: has last-recent data (related to the districts of Barcelona).
- Cloud layer: puts all historical data in their storage environment for any future usage.

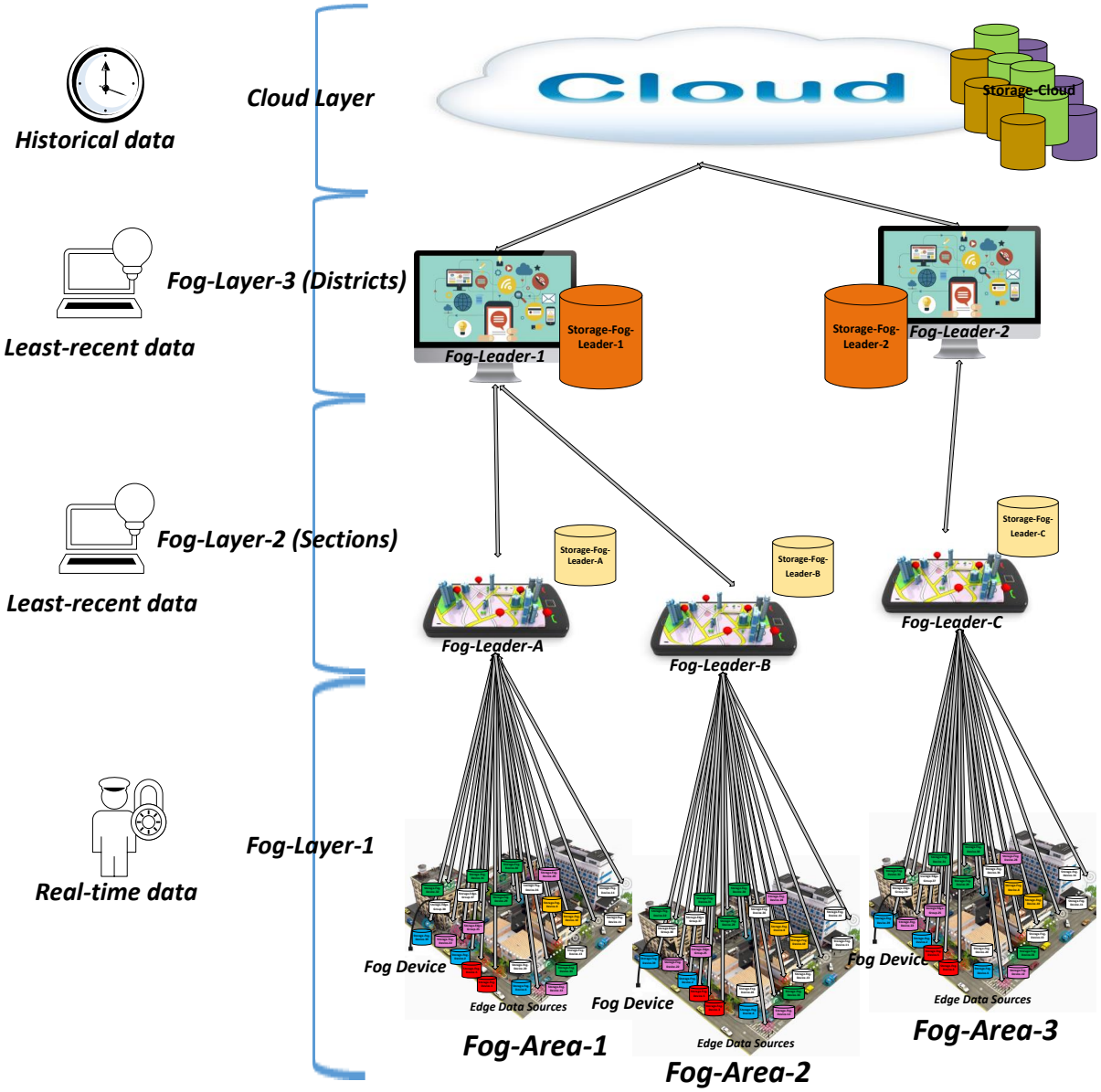


Figure 6.13 Data Storage level in Barcelona through F2C

6.3.2 Estimating Data Storage Size in F2C

As we described in the previous Chapter, we calculated the number of the sensors data production in Barcelona Smart City. So, we show total amount of the data production for each category of information (per day) in Barcelona Smart City as shown blow:

- Energy monitoring category: 2,390,344,704 byte per day.
- Noise monitoring category: 641,280,000 byte per day.
- Garbage Collection category: 360,000,000 byte per day.
- Parking Spot monitoring category: 320,000,000 byte per day.
- Urban Lab monitoring category: 4,723,200,000 byte per day.

Therefore, this amount of produced data must be saved in the different storage layers in the city of Barcelona as shown in Figure 6.13 as shown more details in below.

In this section, we aim to show how much size of our storage at minimum and maximum size in different F2C layers from fog to cloud. And then, we will show how much the total size of the data in the different storages at F2C layers. Therefore, this section is organized as follow: first, we present our methodologies. Second, we show our results. And finally, we discuss about the results.

6.3.2.1 Methodologies

In this subsection, we present our methodologies for how we estimate the minimum and maximum and total size of the storage in F2C layers as shown in below:

1- Estimation of the maximum and minimum data size in the data storage at F2C layers:

We will appraise that the minimum and maximum data size in the data storage at F2C layers. In this way, first we will use the data type with the lowest and highest data production rate. And then, we will take the data type with the lowest and highest data aggregation rate.

2- Estimation of the total produced data in the data storage at F2C layers:

In the previous Chapter, we estimated the total data production numbers. So, the total data production numbers must be saved in each layer of F2C (including Fog-Devices, Fog-Leaders and cloud).

3- Estimation of the data storage efficiency:

As we talk above, we aim to calculate the capacity of the data storage after applying some optimization techniques (such as data aggregation). So, in this case we will present the data storage efficiency rate with respect the data aggregation efficiency rate (as shown in Chapter 5). Therefore, in this section we depict that how much data will be stored in different situation (the normal data volume, the aggregated data volume, the aggregated and compressed data volume) at F2C layers.

6.3.2.2 Results

All below figures in this section follows: the data will be produced by a sensor and then sent to the related Fog-Device which has capability to store the sensor data. And then, Fog-Device will

send the data to the related Fog-Leader (section) in Fog-Layer-2. Similarly, the Fog-Leader (section) will send the data to the related Fog-Leader (district) in Fog-Layer-3. Finally, all districts data will be sent to the data centers in the cloud.

As shown in Figure 6.14, we aim to estimate the minimum data storage size (in terms of the lowest data production rate and highest data aggregation rate) at F2C layers. So, we selected the noise (first type) data type which has the minimum data production rate among all sensors data in Barcelona city (as shown in Figure 5.30). In addition, we chose the "Les Corts" district as a district with minimum sections in the city (as shown in Table 5.3).

Figure 6.14 highlighted the produced data (by a noise sensor) is 768 byte per day. This amount of data (orange color) transferred to the Fog-Device storage. Fog Device storage is able to follow the data aggregation techniques to reduce the size of the data to the 192 byte per day (blue color). Further, the aggregated data can be compressed to reach to the 150 byte per day (gray color). Similarly, we have 105,216 (normal data volume), 26,304 (aggregated data volume), and 20,517 (aggregated and compressed data volume) byte data in the Fog-Leader (sections) at Fog-layer-2. Additionally, we stored 311,040 (normal data volume), 77,760 (aggregated data volume), and 60,653 (aggregated and compressed data volume) byte data in the Fog-Leader (districts) at Fog-layer-3. Indeed, the all stored data (with same size) will be transferred to the data centers in the cloud.

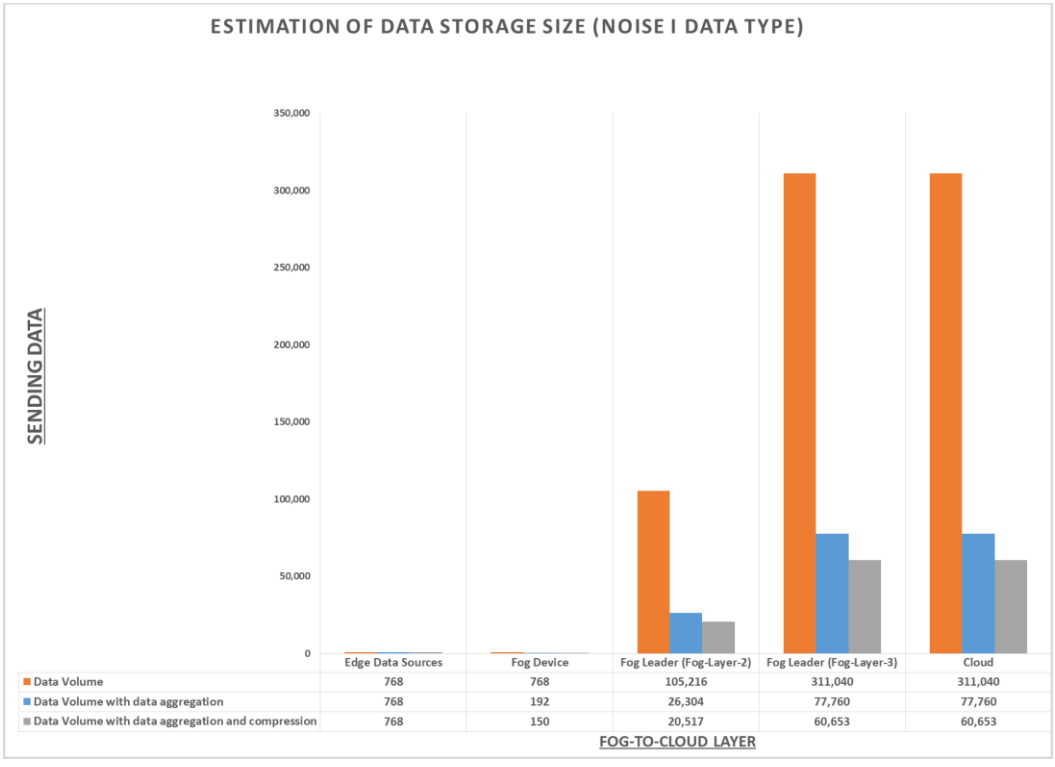


Figure 6.14Estimation of data storage size for the Noise I data type (Noise Category)

As shown in Figure 6.15, we aim to appraise the maximum data storage size (in terms of the highest data production rate and the minimum data aggregation rate) at F2C layers. So, we chose the traffic data type which is in the urban lab category as minimum data production rate among all sensors data in Barcelona city (as shown in Figure 5.30). In addition, we selected the "Nou Barris" district as a district with maximum sections in the city (as shown in Table 5.3).

Figure 6.15 depicted the produced data (by a traffic sensor) is 63,360 byte per day. This amount of data (orange color) sent to the Fog-Device storage. Fog Device storage can provide the data aggregation techniques to reduce the size of the data to the 44,352 byte per day (blue color). In addition, the aggregated data can be compressed to reach to the 34,595 byte per day (gray color). Similarly, we have almost 35 MB (normal data volume), 25 MB (aggregated data volume), and 19 MB (aggregated and compressed data volume) byte data in the Fog-Leader (sections) at Fog-layer-2. Additionally, we saved around453 MB (normal data volume), 317 MB (aggregated data volume), and 247 MB (aggregated and compressed data volume) byte data in the Fog-Leader (districts) at Fog-layer-3. Indeed, the all stored data (with same size) will be sent to the data centers in the cloud.

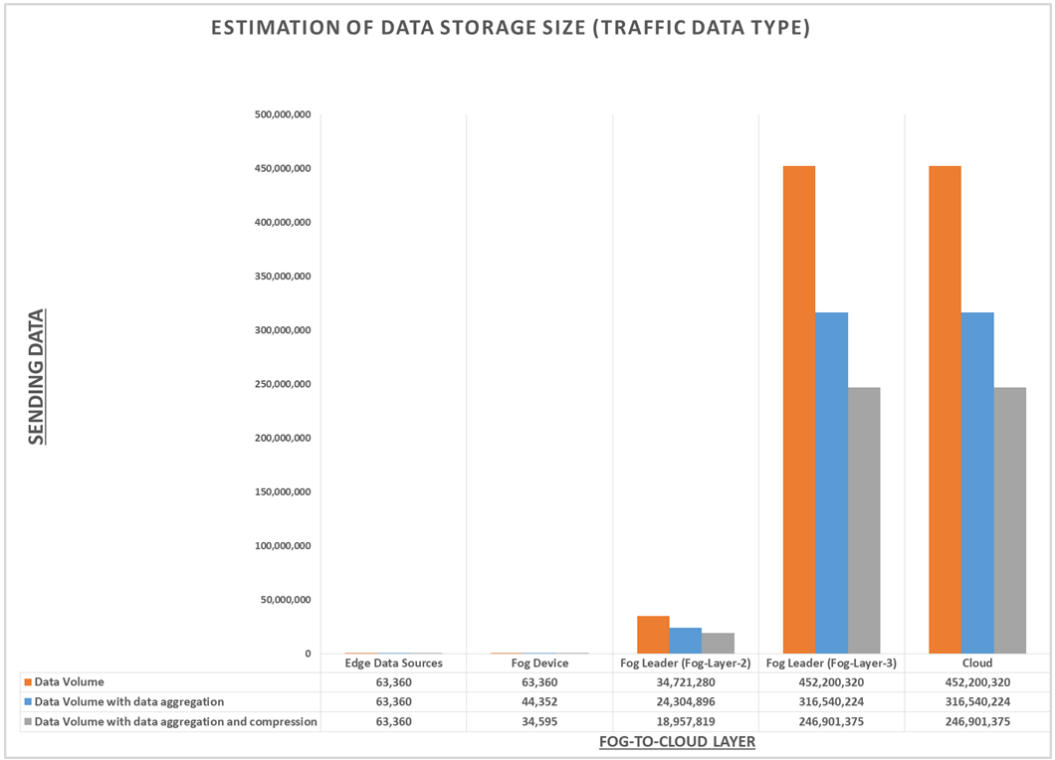


Figure 6.15Estimation of data storage size for the Traffic data type (Urban Lab Category)

Now, we aim to estimate the total size of the data storage at F2C layers. As shown in below figures in this subsection, each figure shows sending data by normal data pushing (orange color), pushing aggregated data (blue color), pushing aggregated and compressed data (green color) through different layers (including Fog-Device, Fog-Leader in Fog-Layer-2), Fog-Leader (inFog-Layer-3), and cloud layer).

- Energy monitoring:

  As shown in Figure 6.16, the storage size for each layer is shown as below:

  - Edge-data-sources layer: this layer generated 2,390MB data per day.
  - Fog-Device layer: this layer stored almost 2,390MB (receiving data), 1,195 MB (receiving data and applying data aggregation), 263 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-2): this layer saved 2,391MB (receiving data), 597 MB (receiving data and applying data aggregation), 132 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-3): this layer recorded around 2,391 MB (receiving data), 299 MB (receiving data and applying data aggregation), 66 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Cloud layer: this layer received same amount of data from Fog-Leader (in Fog-Layer-3) layer. This layer provided almost 2,391 MB (receiving data), 299 MB (receiving data and applying data aggregation), 66 MB (receiving data and applying data aggregation and compression) byte for saving data per day.

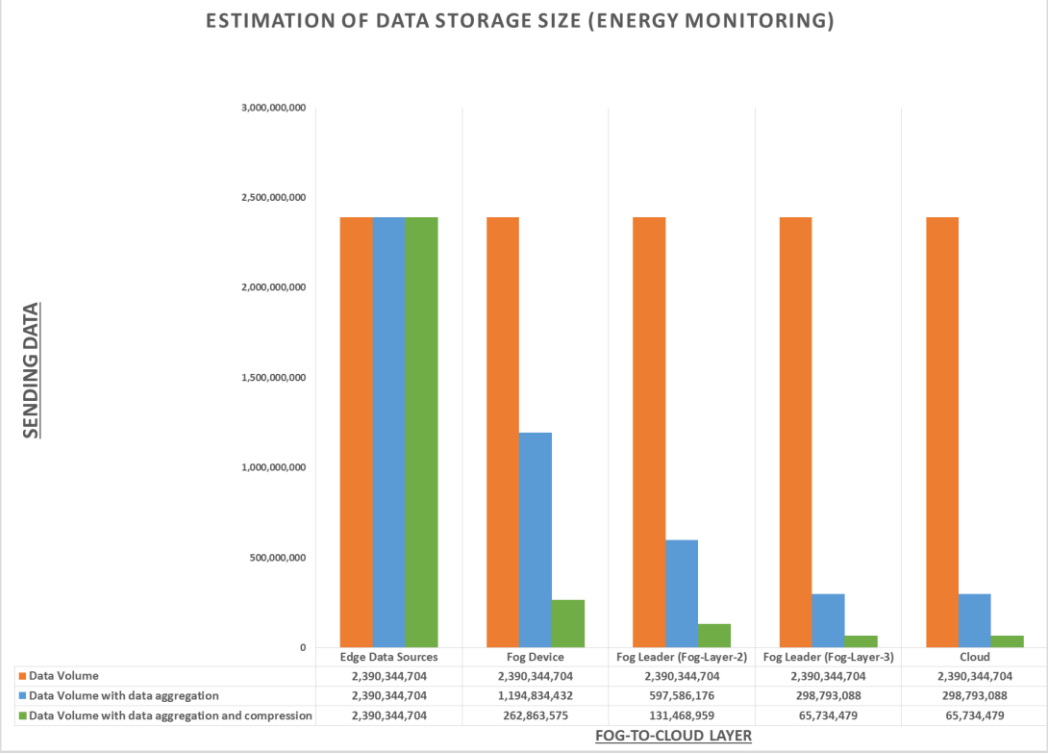


Figure 6.16 Estimation of data storage size (Energy Monitoring)

- Noise monitoring:

  As shown in Figure 6.17, the total data storage size (including sending data, data with aggregation and data with aggregation and compression) for the noise monitoring category as shown as below:

  - Edge-data-sources layer: this layer generated 642 MB data per day approximately.
  - Fog-Device layer: this layer saved almost 641 MB (receiving data), 160 MB (receiving data and applying data aggregation), 35 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-2): this layer stored around 642 MB (receiving data), 39 MB (receiving data and applying data aggregation), 8 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-3): this layer registered641 MB (receiving data), 9 MB (receiving data and applying data aggregation), 2 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Cloud layer: Similarly, this layer took same amount of data from Fog-Leader (in Fog-Layer-3) layer. This layer saved641 MB (receiving data), 9 MB(receiving data and applying data aggregation), 2 MB (receiving data and applying data aggregation and compression) byte for saving data per day.

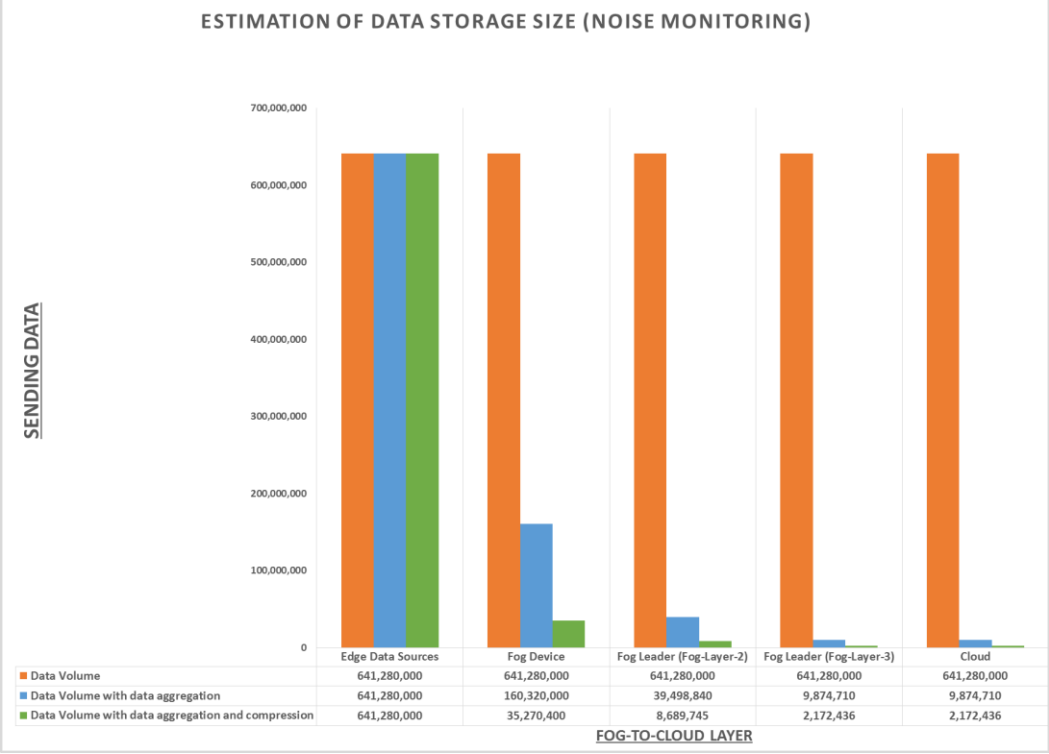


Figure 6.17 Estimation of data storage size (Noise Monitoring)

- Garbage Collection:

  In Figure 6.18, the garbage collection category must have the specific total data storage size (including sending data, data with aggregation and data with aggregation and compression) for each layer as shown as below:

  - Edge-data-sources layer: this layer produced 360 MB data per day.
  - Fog-Device layer: this layer stored around 360 MB (receiving data), 108 MB (receiving data and applying data aggregation), 24 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-2): this layer saved almost 360 MB (receiving data), 33 MB (receiving data and applying data aggregation), 7 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-3): this layer registered 360 MB (receiving data), 9 MB (receiving data and applying data aggregation), 2 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Cloud layer: this layer received same amount of data for saving in the storage from Fog-Leader (in Fog-Layer-3) layer. This layer recorded 360 MB (receiving data), 9 MB (receiving data and applying data aggregation), 2 MB (receiving data and applying data aggregation and compression) byte for saving data per day.

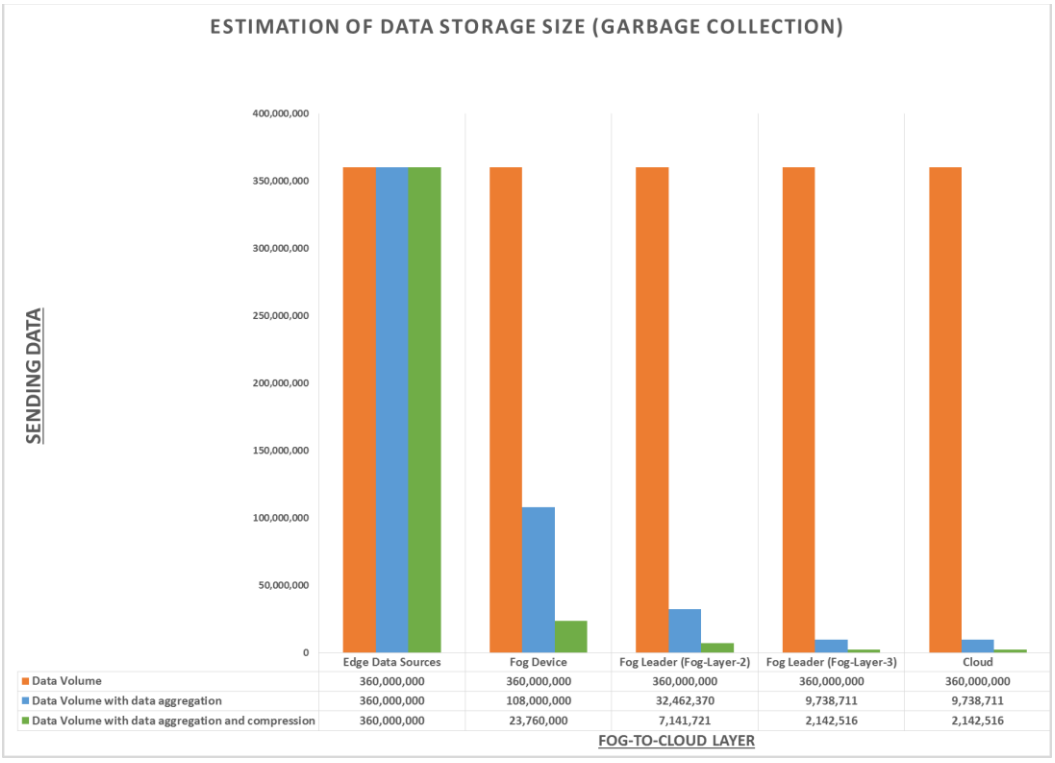
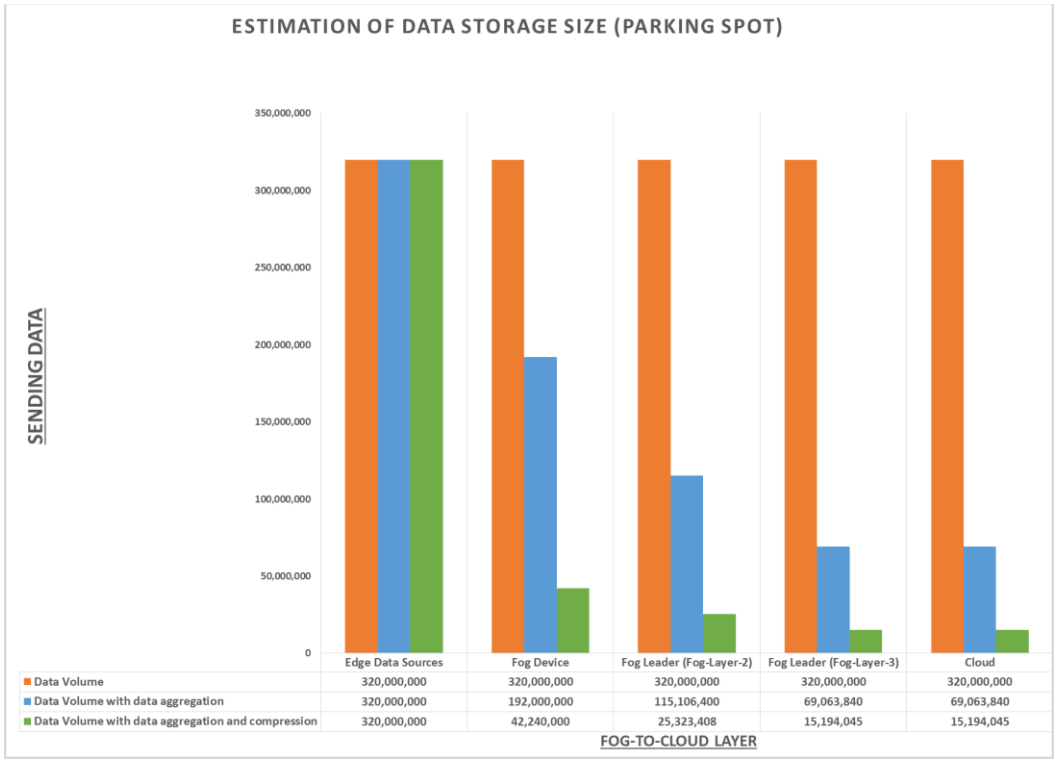


Figure 6.18 Estimation of data storage size (Garbage Collection)

- Parking Spot:

  In Figure 6.19, the parking spot category must provide the specific amount of storage size for total received data in each layer as shown as below:

  - Edge-data-sources layer: this layer produced 320MB data per day.
  - Fog-Device layer: this layer stored around 320 MB (receiving data), 192 MB (receiving data and applying data aggregation), 42 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-2): this layer saved almost 320 MB(receiving data), 116 MB (receiving data and applying data aggregation), 26MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-3): this layer registered 320 MB (receiving data), 69 MB (receiving data and applying data aggregation), 16 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Cloud layer: this layer received same amount of data for saving in the storage from Fog-Leader (in Fog-Layer-3) layer. This layer recorded 320 MB (receiving data), 69 MB (receiving data and applying data aggregation), 16 MB (receiving data and applying data aggregation and compression) byte for saving data per day.



	Edge Data Sources	Fog Device	Fog Leader (Fog-Layer-2)	Fog Leader (Fog-Layer-3)	Cloud
Data Volume	320,000,000	320,000,000	320,000,000	320,000,000	320,000,000
Data Volume with data aggregation	320,000,000	192,000,000	115,106,400	69,063,840	69,063,840
Data Volume with data aggregation and compression	320,000,000	42,240,000	25,323,408	15,194,045	15,194,045

Figure 6.19 Estimation of data storage size (Parking Spot)

- Urban Lab:

  In Figure 6.20, we must prepare the below total data storage size for the urban category of information:

  - Edge-data-sources layer: this layer generated4,723MB data per day.
  - Fog-Device layer: this layer saved 4,723 MB (receiving data), 3,306 MB (receiving data and applying data aggregation), 727 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-2): this layer stored4,723 MB (receiving data), 2,318 MB (receiving data and applying data aggregation), 510 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Fog-Leader (in Fog-Layer-3): this layer recorded4,723 MB (receiving data), 1,623 MB (receiving data and applying data aggregation), 357 MB (receiving data and applying data aggregation and compression) byte data per day.
  - Cloud layer: this layer recorded same amount of data for storing in the storage from Fog-Leader (in Fog-Layer-3) layer. This layer saved around 4,723 MB(receiving data), 1,623 MB (receiving data and applying data aggregation), 357 MB (receiving data and applying data aggregation and compression) byte for saving data per day.

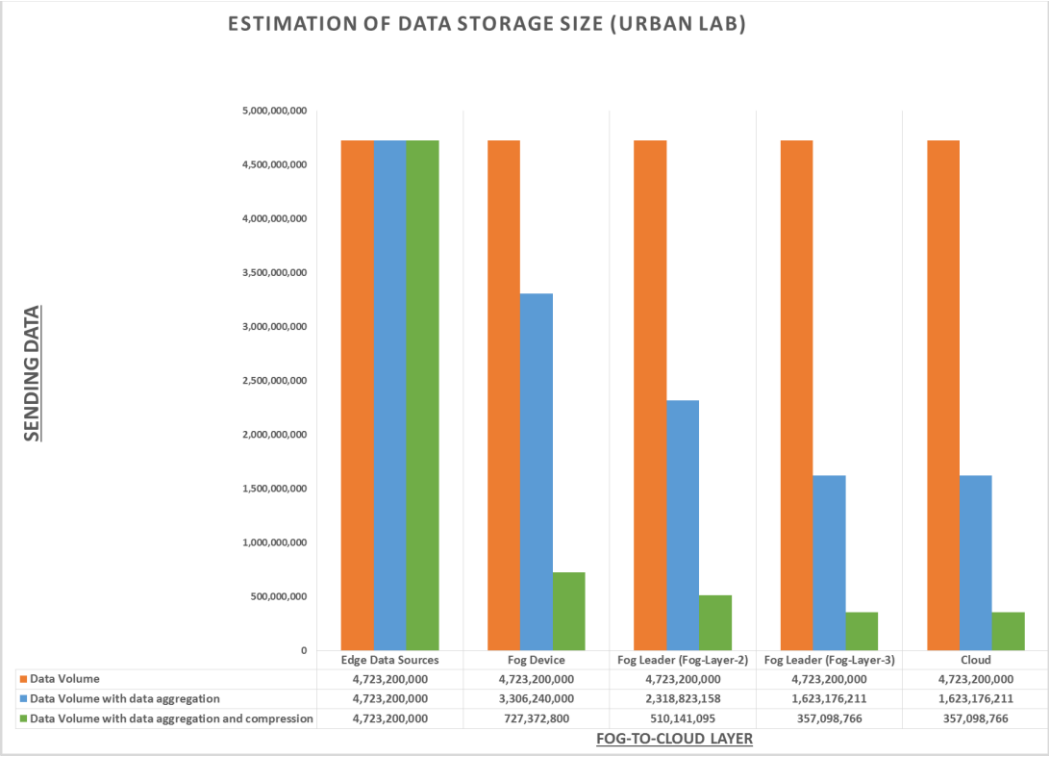


Figure 6.20 Estimation of data storage size (Urban Lab)

6.3.3 Discussion of results

- We observed that the minimum data storage size must be 768 byte at Fog-Device, 105,216 byte at Fog-Leader (Section), and 311,040 byte per day at Fog-Leader (District) in Barcelona Smart City. However, this amount can be organized to move between layers through the specific frequency of update in F2C layers.
- We realize that maximum data storage size must be 63,360 byte at Fog-Device, 35 MB at Fog-Leader (Section), and 453 MB at Fog-Leader (District) per day in Barcelona Smart City. In addition, this amount can be managed to move between layers through the specific frequency of update in F2C layers.
- The data aggregation and compression techniques can get helps to reduce the amount of data in the data storage at each layer of F2C.
- We estimate that total data storage size in maximum situation (for urban lab category) is as shown in below:
    - Fog-Device: 4,723MB data per day in normal data volume generation, 3,306 MB data per day (receiving data and applying data aggregation) and727 MB data per day (receiving data and applying data aggregation and compression).
    - Fog-Leader (section): 4,723 MB data per day in normal data volume generation,2,318 MB data per day (receiving data and applying data aggregation) and 510 MB data per day (receiving data and applying data aggregation and compression).
    - Fog-Leader (districts): 4,723MB data per day in normal data volume generation, 1,623 MB data per day (receiving data and applying data aggregation) and357 MB data per day (receiving data and applying data aggregation and compression).
    - Data centers (cloud layer):4,723MB data per day in normal data volume generation, 1,623 MB data per day (receiving data and applying data aggregation) and357 MB data per day (receiving data and applying data aggregation and compression).
- We realize that total data storage size in minimum situation (for parking spot category) is as described in below:
    - Fog-Device: 320 MB data per day in normal data volume generation, 192 MB data per day (receiving data and applying data aggregation) and42 MB data per day (receiving data and applying data aggregation and compression).
    - Fog-Leader (section): 320 MB data per day in normal data volume generation, 116 MB data per day (receiving data and applying data aggregation) and26 MB data per day (receiving data and applying data aggregation and compression).

- Fog-Leader (districts): 320 MB data per day in normal data volume generation, 69 MB data per day (receiving data and applying data aggregation) and16 MB data per day (receiving data and applying data aggregation and compression).
- Data centers (cloud layer):320 MB data per day in normal data volume generation, 69 MB data per day (receiving data and applying data aggregation) and16 MB data per day (receiving data and applying data aggregation and compression).

6.4Summary and contributions

In this chapter, we make a survey about the Data Preservation block and their related phases (including data classification, data storage, and data dissemination). We said about the definitions, responsibility, advantages, and challenges each phase. Indeed, we also mentioned that there is no any consideration about the Data Preservation block through F2C in Smart City scenario.

Then, we proposed the Data Preservation block and their related phases through F2C for Smart City scenario. We showed that the Fog-Layer-1 is only able to provide the basic level of data actions for all phases of the Data Preservation block. Plus, the Fog-Layer-2 is responsible to apply more sophisticated data actions for all phases of the Data Preservation block. And then, the cloud layer prepares the advanced level of data actions for all phases in this block.

And, we discussed about data storage levels (at Fog-Devices, Fog-Leader, and cloud), available types of the data stored (consisting of real-time, last-recent, and historical data), updating mechanism (from down to upper layers), naming mechanism (with respect to CDN), and an object stored services (including writing scheme for a current sensors data, a new sensor to current Fog-Device, a new Fog-Device to Fog-Layer-1) through F2C for the Data Preservation block in Smart City. So, all the above discussion got helps to handle the data to classify, store, and disseminate through F2C architecture.

Next, we presented the data storage level in Barcelona Smart City. So, we estimated the numbers of data which is saved in Fog-Devices, Fog-Leader (related to the sections of Barcelona), and Fog-Leader (related to the districts of Barcelona). And then, we showed the efficiency rate of applying data aggregation and data compression techniques through F2C at data storage levels in Barcelona.

We have listed the main contributions of this Chapter as shown in below:

- To illustrate the Data Preservation block and their related phases to handle the preservation system through F2C in the Smart City[138].
- To provide facility to store and get access to the real-time, last-recent, and historical data through our F2C data management in Smart City.

- To present the writing schema for an object store service model (as part of the distributed file system) for F2C data management architecture.
- To show a naming mechanism to get access to the data in the Smart City (including three main dimensions which are data type, location, and time).
- To introduce the updating frequency mechanism to send real-time to upper layer after the defined time.
- To estimate the capacity of the data stored in each layer of F2C data management architecture in Barcelona Smart City.

The data preservation block (through our F2C data management architecture) makes several advantages as shown in below:

- To have access to the real-time, last-recent, and historical data through our F2C data management in Smart City.
- To present the writing schema for an object store service model (as part of the distributed file system) for F2C data management architecture.
- To organize the updating frequency mechanism through city manager policies and business models (for real-time, last-recent, and historical data).

We published different publications for this Chapter in the reputable venues as shown in the below:

- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, X.Yin, C.Wang, "A Data LifeCycle Model for Smart Cities", IEEE conference on ICTC 2016, Korea, October 2016.

Chapter 7:

The Data Processing Block

On one hand, most of the related work mentioned that cloud computing is responsible to handle data processing for all collected data in Smart City [152, 159]. On the other hand, fog computing provides the facility to demonstrate data processing at any edge layer, according to the requirements of the application or service.

In this Chapter, we aim to present the following concepts: In section 7.1, we show the phases in the Data Processing block. We also discuss all definition, state of the art, and objectives of each phase. In Section 7.2, we argue about the Data Processing block in F2C. Then, we introduce the proposal of all phases in the Data Processing block (including data process and data analysis). In section 7.3, we conclude the main contributions of this Chapter.

7.1 Phases in the Data Processing Block

Data Processing block is able to provide a subset of processing techniques to convert raw data to information which provides the facility to be used by different services. There are many existing references which mention that data processing mainly is acquired in the cloud computing environment in different scenarios such as smart cities [152, 159].

In our point of view, the Data Processing block is responsible for performing the main big data processing, extracting knowledge or generating additional value, through sophisticated data analysis techniques. The results of the processed data (higher value data) can be delivered to the end users, or stored for future additional data reuse or reprocessing.

As shown in Figure 7.1, Data Processing block includes with Data Process and Data Analysis phases which will each be described below.

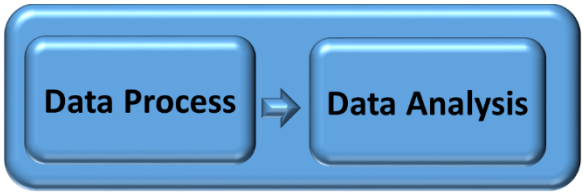


Figure 7.1Phases in the Data Processing Block

The most important objectives of this block, are as shown as below:

- To provide a set of processes for the raw data to convert the data to meaningful information.
- To get help for any future usage of the processed data by applying some appropriate data analysis and analytic techniques.

7.1.1 The Data Process phase

The Data Process phase is the first phase in the Data Processing block as shown in Figure 7.2. We will discuss more about this phase as the following sentences.

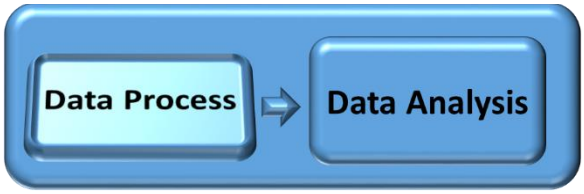


Figure 7.2 The Data Process phase in the Data Processing Block

7.1.1.1 Definition

Data process phase must be able to handle a sub set of processes in the system. Those processes must be able to cover all user requirements through the related services. In addition, a bunch of processing techniques (including parallel processing) must be able to provide such kind of initial feeds for several applications.

7.1.1.2 State of the art

As we mentioned above, many authors highlighted that cloud computing is a place to demonstrate data processing for all collected data in any big data environment such as a Smart City[152, 159] in one side. On the other side, in [163] said that there are three different types of process model which we can apply to any big data environments which are generic processing, graph processing, and the stream processing model. In addition, in [190], the author designed the BigETL for Smart City data processing to provide a high flexibility for processing different types, formats, and sizes of the data. The BigETL consists of multiple data processing systems in its underlying layer (supporting Spark, Hive, Linux Shell, SQL Engine, and Python environment).

In[191], the author proposed an object store service model to read data from edge sources to provide data for services. As mentioned in the data preservation block, the author says that there are two different protocols to get an object store service (IPFS and Scale-out NAS) from edge to fog computing. First, IPFS is able to read the object through local stored data node or metadata management in DHT (located in Fog computing). However, there are some challenges to read an object with data stored locally through accessing the DHT in terms of accessing time and reading data from the local data stored. So, the second model is comprised of the combination of DHT and Scale-out NAS. This combination model can handle to read the data from edge to fog node more efficiently in terms of traffic and time.

In our point of view, the Data Process phase provides a set of processes to transform (raw) data into more sophisticated data/information. These processes could include one or several internal steps, such as preprocessing or post-processing, depending on the particular business

requirements. Data considered for processing can be either real-time, just generated, data (from the Data Acquisition block), or historical archived data (from the Data Preservation block). The output of this phase is considered higher value data, meaning that this data is more mature than the original (raw) input data.

7.1.1.3 Objectives and challenges of an effective Data Process phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data process phase as part of a data management architecture. The main objectives and challenges are the following:

- Performing all data processing required in the application or service to convert raw data into some more sophisticated, higher level information, which provide smartness to the service. These processes could include one or several internal steps, such as preprocessing or post-processing, depending on the particular applications requirements.
- Performing a set of data processing techniques which must be matched with the user requirements and business models.
- These objectives contribute in simplifying the data challenges of :
    - Data Volume, Data Variety, and Data Veracity: because Smart City data (comes with abundant heterogeneous sources and data types and formats) imposed difficulties for data processing (including data with different level of qualities and different types of sensitive information) [190].
    - Data Value: Grabbing value among large amounts of data by the set of efficient processing.
    - Data Velocity: Data is produced quickly. So, applying a set of efficient data processing are challenging concepts through distributed environments.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. We just present how the services can look for the required data(through an F2C object store service model) in F2C layers (traditionally most of service developer builds their services and applications through cloud environments). This will be described and discussed in subsection 7.2.

7.1.2The Data Analysis phase

The last phase of the Data Processing block is Data Analysis as shown in Figure 7.3.

Figure 7.3 The Data Analysis phase in the Data Processing Block

7.1.2.1 Definition

Big data paradigm imposes high speed generation of large amounts of data which we need to design data analysis techniques to discover insights of data [218, 219]. In a cloud computing environment, data analysis (consisting of different analytical techniques and methods) considers to be started their tasks after data processing [218].

7.1.2.2 State of the art

There are several study to show the performance of data analysis in different scenarios. In cloud computing, there are three main techniques to overcome data analysis which are statistical analysis (including specific models for predication and summarizing datasets), data mining (consisting of a variety of techniques, clustering, classification, etc., to explore patterns and models present in the data), and machine learning (discovering relationships that are present within the data) [218]. In addition, [220] mentioned that a huge amount of data can be processed over very large clusters through data analysis techniques in cloud computing. Recently, there is some effort to handle big data analytics in the fog layer. So, this work proposed that fog engines can handle data analytics in the edge of networks [23].

In our point of view, the Data Analysis phase is responsible for developing all data analysis and data analytics for extracting knowledge and discovering new insights. This phase is the last step in the procedure of value generation, and it is usually the natural interface with the end-user. Alternatively, this data can also be considered for archiving and stored, as part of the Data Preservation block.

7.1.2.3 Objectives and challenges of an effective Data Analysis phase

According to the reviewed literature, a number of objectives and challenges can be defined in order to design an efficient data analysis phase as part of a data management architecture. The main objectives and challenges are the following:

- Performing all deep data analysis and data analytics algorithms for extracting knowledge and discovering new insights. Again, the analysis or analytics processes tightly depend on the users' application or business models.
- This phase also provides a user interface for accessing the results of data processing of an application or service.
- Providing analysis and analytics techniques to estimate and predict the behavior of the systems and datasets.
- These objectives contribute in simplifying the data challenges of :
    - Data Volume: because the huge number of data must be analyzed in the distributed environments.
    - Data Variety: because the efficient data analyses and analytics that must be applied to each data have a different type and format.
    - Data Value: Extracting value among the big amount of data through data analysis and analytics techniques.

Our distributed hierarchical F2C data management architecture can effectively include and address most of the objectives and challenges listed above. This will be described and discussed in subsection 7.2.

7.2The Data Processing Block in F2CSmart City

Data processing can be performed at any F2C layer, according to the requirements of the application or service. For instance, critical real-time services will be executed at fog layer 1 in order to have a faster access to the (just generated) real-time data. Note that accessing data locally inside the boundaries of a fog node is much faster than moving the data to a centralized cloud data center and, after, reading these same data from the cloud to the local node.

Alternatively, deep computing complex applications will be executed at the cloud layer. Note that i) in the cloud the computing resources are unlimited and, ii) the data set of a high performance computing application will presumably be very large and, therefore, be part of the historical data set stored at the cloud layer. Note that in this case, where computation requires very high capabilities, adding more latency to the first access to data will not be significant in the overall performance.

For the other applications, they will be executed at the lowest fog layer that provides the required computing capabilities and the lowest fog layer that contains the required data set. As a general rule, the closer the layer, the faster the responses times. An additional consideration in this case is when the required data is not present in the current fog node at layer 1, but can be accessed from either a node at a higher layer or a neighbor fog node at the same layer 1. This option may

eventually be considered and solved using some sort of cost model to estimate the effects of both cases and proceed according to the lowest cost.

The F2C model provides facilities to organize all phases of the data processing block from the fog to the cloud layers as shown in Figure 7.4. We have a bit sophisticated process stage in the Fog-Layer-2. And finally, the cloud provides the complete processing stage.

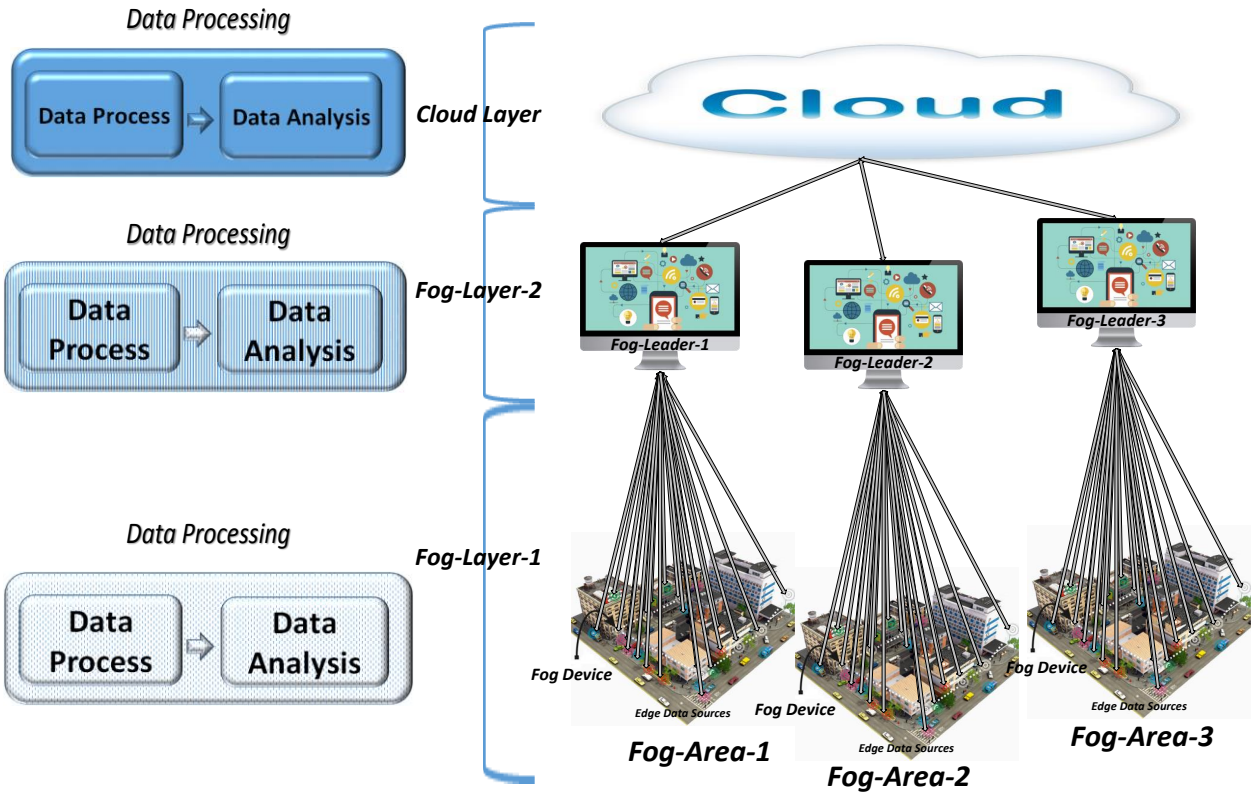


Figure 7.4 Description Scenario of Data Processing Block

7.2.1 An F2C object Store Service model for F2C computing model

As we described in the previous chapter, an object store service model is responsible for organizing the data location in an object store. So, the read concepts (as part of an object store service) provide the facility to find data objects in our hierarchal distributed model through F2C computing which will be detailed further in this section.

As you see in the previous chapter, the scenario is described. So, in this section, we depict our reading scheme to find the location of our object through F2C data management architecture. And then, we concentrate on how the reading scheme will be handled in each layer of Fog-Layer-1, Fog-Layer-2, and cloud computing as shown below.

7.2.1.1 Read the requested data from DHT in Fog-Layer-2

As shown in Figure 7.5, the first case is that the client is located in the edge-data-sources layer and requests for data to Fog-Layer-2. So, the request will be transferred to Fog-Layer-2. And then, the related DHT (of the Fog-Leader-2) looks for the requested data. In this case, the path of stored data will be found in the DHT. So, the DHT will find the data in the related data storage. And then, the data will be sent to the client.

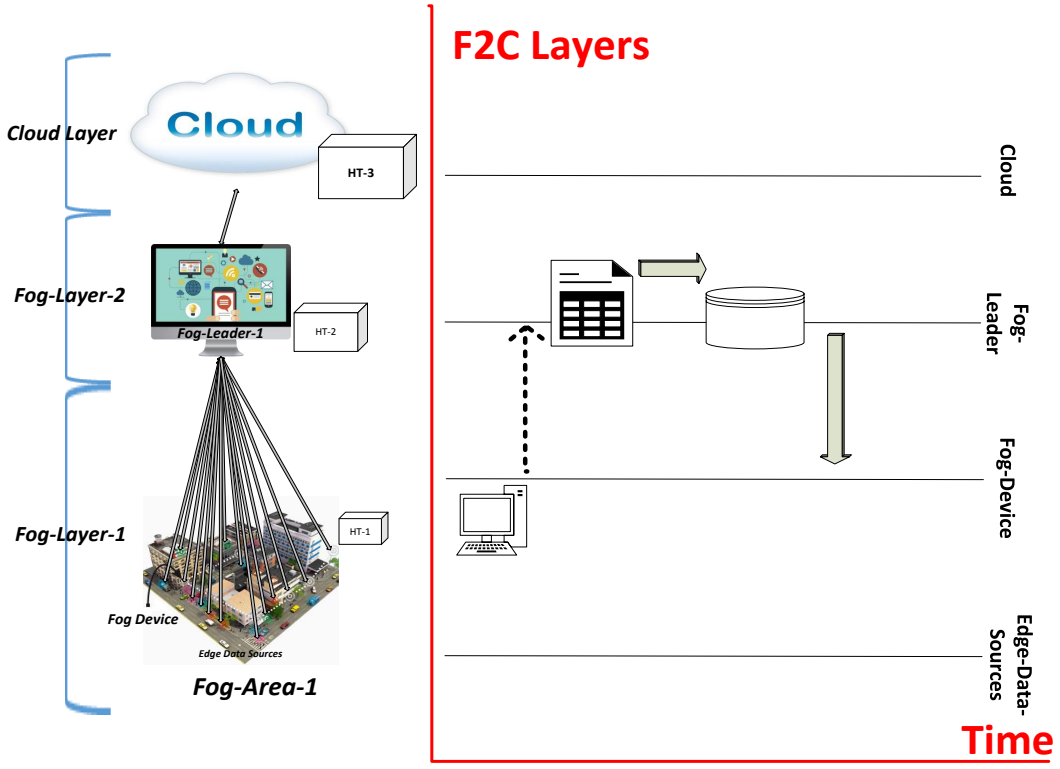


Figure 7.5Reading schema for an object Store Service model (Fog-Layer-2)

7.2.1.2 Read the requested data from DHT in Cloud

Again, the client will send a request for collecting data from edge-data-source. This request will pass to closet Fog-Layer-2 as shown in Figure 7.6 Then, the request data will not exist in the related DHT (in Fog-Layer-2). So, in this case, the message will be transferred to cloud layer. The cloud will look for the data in the DHT. Obviously, the cloud can find the path of the requested data because the cloud has all data path information in their glossaries. Indeed, the data will be found in the related storage and sent to the client.

In addition, if in case the requested data cannot be found in the cloud layer as shown in Figure 7.7, the cloud sends a message to the client to say that this data cannot exist in this F2C data management architecture.

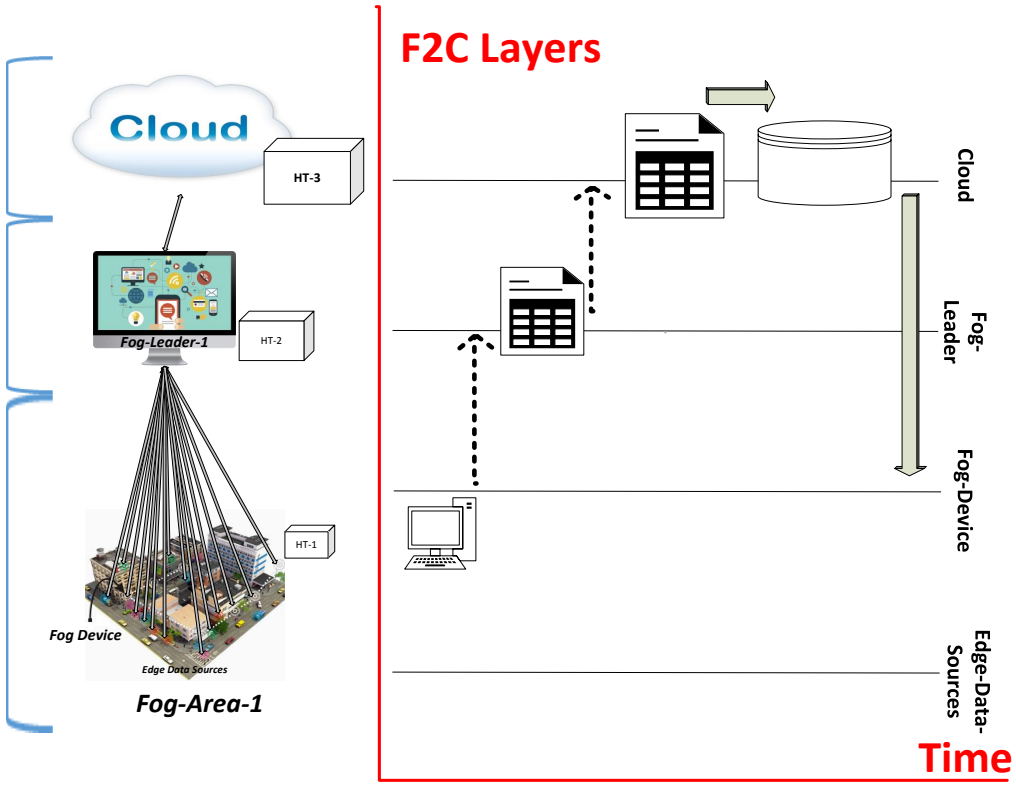


Figure 7.6 Reading schema for an object Store Service model (cloud layer)

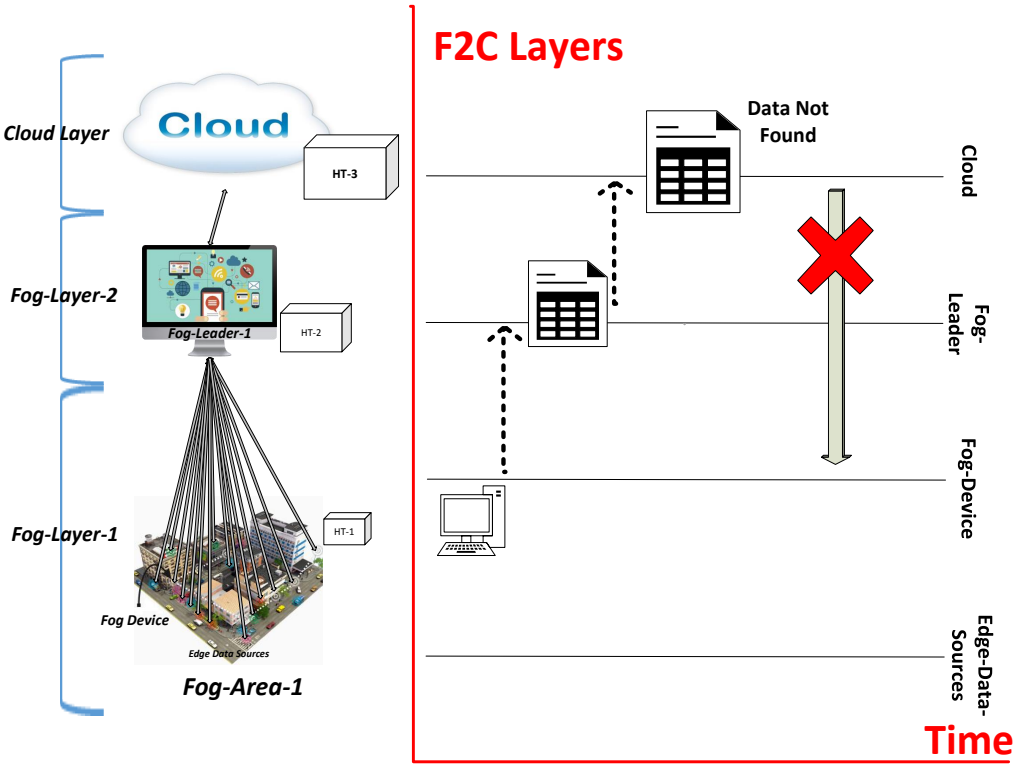


Figure 7.7 Reading schema for an object Store Service model (Not found in the cloud layer)

7.2.1.3 Estimation of reading data through layers of F2C

There are different ways to connect Edge-Data-Sources, Fog Device, Fog-Leader, and cloud components to each other through different types of network communications (including wired and wireless connections) regarding the potential of the smart cities. In fact, services will be run somewhere in our F2C data management architecture to get their appropriate data for their future purposes. So, it is necessary to estimate that network latency rate to find out the suitable path to get the data for the services.

In Figure 7.8, we proposed the different rate of the network latency rate between F2C layers (including from edge-data-sources to Fog-device, Fog-device to Fog-Leader, Fog-Leader to cloud layer). In fact, network latency shows any kind of delay that acquires in data communication over a network. We called the network a low-latency network when we have small rate of delay in our network. Otherwise, we say the network has high-latency over their network components.

We observe that the network latency rate can be considered with the specific rate between layers as shown in more detail below. In addition, the base of this proposed network latency rate is in[191]:

- The one-way network latency rate from one node (such as temperature sensor) in the edge-data-source layer to another node (such as noise sensor) in the edge-data-source layer: L(Fog)=10~100ms;
- The one-way network latency rate from one node in the edge-data-source layer to Fog-Device layer: L(Fog)=10~100ms;
- The one-way network latency rate from one Fog-Device layer to Fog-Leader: L(Core) =50~100ms;
- The one-way network latency rate from one Fog-Leader node to cloud layer: L(Cloud) =100ms.In addition, this number is variable and unpredictable in most of the times.

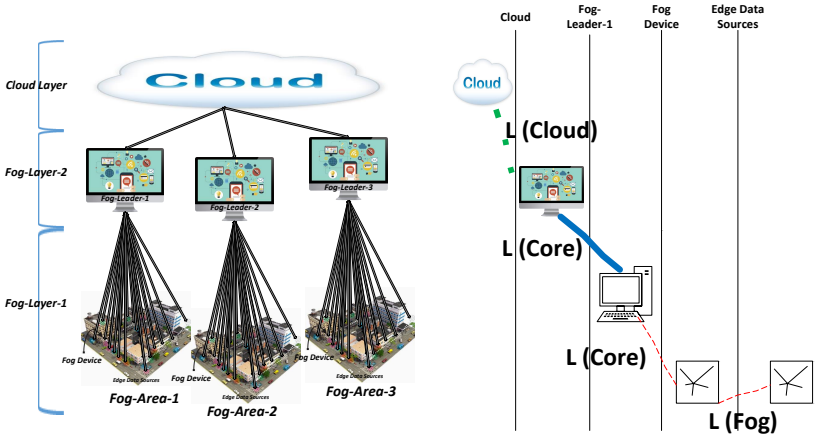


Figure 7.8The one-way network latency rate in F2C data management architecture.

7.3Summary and contributions

In this Chapter, we presented a survey about the Data Processing block and their related phases (including data process and data analysis). We discussed in detail about the definitions, state of the art, responsibility, advantages, and challenges of each phase.

We then depicted the Data Processing block and their related phases through F2C for Smart City scenario. We showed that the Fog-Layer-1 only covers the basic level of data actions for all phases of the Data Processing block. Then, the Fog-Layer-2 is able to handle more sophisticated data actions for all phases of the Data Processing block. And, the cloud layer provides the advanced level of data actions for all phases in this block.

Next, we mentioned the main concepts of an F2C object store service model for the reading scheme in the Data Processing block. We saw that there are different aspects of the F2C object store service model which includes reading data from different layers of the distributed hierarchy architecture in the smart cities. We showed all different scheme models by their related pictures in the aforementioned sentences.

Indeed, we introduced the estimation of the reading data through layers of F2C. This means that we showed how much network latencies we have in the layers of F2C. However, some of these numbers are unpredictable and very dependent on the network communication architectures and the urban structure of the city.

We have listed the main contributions of this Chapter as shown below:

- We proposed the processing block (including data process, and data analysis phases) for F2C data management architecture [208].
- We made the survey about definition, state of the art, and challenges for the data processing block (including data classification, data storage, and data dissemination phases).
- We generated a reading schema for an object Store Service model (distributed file system) for F2C data management architecture.
- Proposing the one-way network latency rate in the layers of the F2C data management.

The Data Processing block (through our F2C data management architecture) can provide some desirable advantages as described below:

- There is the possibility to build the different types of services (with real-time, last-recent, and historical data access) through data processing block in F2C data management architecture.
- Providing an effective interface to get easy access to the related data for the services (including three main dimensions which are data type, location, and time).

We proposed some publications for this Chapter in the reputable venues which can be sorted as below:

- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, X.Yin, C.Wang, "A Data LifeCycle Model for Smart Cities", IEEE conference on ICTC 2016, Korea, October 2016.
- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera,"A Novel Architecture for Efficient Fog to Cloud Data Management in Smart Cities", IEEE ICDCS 2017, Atlanta, USA, June 2017.
- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera,"Fog to Cloud Data Management in Smart Cities", IEEE FTC 2017, Vancouver, Canada, November 2017.

Chapter 8:

Conclusions and Future work

The amount of information to be generated in a smart city (as an example of Big Data environment) is very high, but it will be even much higher in a near future. In fact, it is no doubt that data is becoming a valuable asset in the recent decades. Hence, the society as a whole is encouraged to use this asset for enriching its day-to-day activities, through more efficient and smart solutions in the smart cities. Furthermore, the strong data growth rate, the new and high demanding data stakeholders, and the new innovations and technologies are ascertaining many new challenges and complexities for data life management, from creation to consumption, in terms of different data formats always considering the enormous volume of data to be managed in the smart cities scenarios.

In this thesis, we have shown the complexity of data management and highlighted the importance of Data LifeCycle (DLC) models as the initial point for designing an efficient and comprehensive data management architecture. For this reason, we have initially surveyed most existing DLC models proposed to be a high level solution to manage data life from production to usage. The result of this evaluation shows that there is no comprehensive DLC model to manage the data life from its production up to its consumption in today's data world. So, we have presented the COmprehensive Scenario Agnostic DLC (COSA-DLC) model, and demonstrated its completeness with respect to the 6Vs challenges. This model is abstract, as it has not been designed for any specific scenario; however, it can be easily adapted to any particular scenario. As a follow up of this work, we have adapted the COSA-DLC model to a Smart City scenario with complex and flexible data management requirements. The new model has been named the Smart City Comprehensive DLC (SCC-DLC) model, and it is the seed of our data management architecture design. The advantages of our SCC-DLC model can be summarized as:

- Can be applied to any Smart Cities scenario easily
- Covers 6Vs challenges as a widely accepted concept of Big Data
- Provides a comprehensive model for data management in Smart Cities
- Gives some clues to developers and managers of Smart Cities to handle their enquiries regarding data life and stages (from creation to consumption)
- Organizing and managing data without any limitation about hardware and software
- Making facility to have standardization and globalization for the Data management model in the Smart Cities.
- The model considers data during their whole data life cycles, from production to consumption and cleaning, including storage and processing.

Next, we have presented a novel architecture for hierarchal distributed data management in smart cities based on a distributed hierarchical Fog to Cloud (F2C) resources management system. The F2C data management architecture is organized in three main blocks: the data acquisition, the data preservation and the data processing. For each block, we have described the main functionalities and defined the main objectives and challenges that must be addressed in order to

implement them in a smart city scenario. The advantages of this data management architecture are numerous, and can be summarized as:

- This architecture can benefit from the combined advantages of both, the cloud and the fog computing technologies, these are high computing and storage capabilities from the cloud layer and reduced network traffic and communication latencies from the fog layers.
- Real-time data accesses are much faster than in a centralized architecture. This higher speed is not only due to the reduced communication latencies of proximity, but due to the fact that accessing data from a centralized system requires the data to be moved first to the cloud, classified and stored there, and then moved back to the edge. So two times data transfer through the same path.
- By reducing the data transmission length, the security risks and the probability of communication failure are both reduced and, additionally, privacy can be easily enhanced.
- By having the just collected data available at fog layer 1, the network load is drastically reduced because some applications will be able to access these data locally, avoiding several remote data accesses through the network.
- By having the just collected data available at fog layer 1, the transmission to the cloud is not urgent and, therefore, it can be delayed without any performance loss. This allows additional optimization implementations, such as:
    - Performing some data aggregation techniques to reduce the volume of data to be transmitted upwards, without any computational constraint, as data do not need to be sent immediately.
    - Adjusting the frequency of the data transmission in order to use the network in periods when the traffic load is low.
- Traditional centralized systems define a low-frequency policy for data collection from sensors in order to reduce the total amount of data to be transmitted in the network. By having the real-time data available at fog layer 1, the data collection frequency can be increased at this level without overloading the network and, therefore, provide more precision and accuracy from the sensed data at no additional cost.
- By defining a distributed storage hierarchy, data can be cached at different layers of the architecture and, for this reason, data access times can be easily reduced.
- In addition, the F2C architecture can manage data according to their initial location, which enables exploiting some locality features required in Smart City IoT contexts.
- From the processing and analytics point of view, this architecture allows a flexible interface in order to access the most convenient data for each service or application in an IoT context.
- During processing time, the architecture hierarchy provides an efficient structure that allows the application to access the nearest (and therefore fastest) data from the original data source.

- And finally, thanks to the distributed nature of the F2C data management model, it allows performing additional data related optimizations, such as providing high levels of quality, keeping high security and privacy standards, as well as reducing the global network performance.

The most obvious advantage is that high computing and storage capabilities from the cloud layer can be combined with reduced network traffic and communication latencies from the fog layers, while enhancing fault tolerance and security and privacy protection. However, by providing such a hierarchical and distributed model, some interesting additional advantages arise:

- Real-time data accesses are much faster than in a centralized architecture;
- The network load is drastically reduced because many data can be accessed and used locally;
- Several aggregation techniques can easily be applied to further reduce the volume of data to be transferred through the network;
- The data transmission frequency can be adjusted in order to use the network in periods of low traffic;
- The data collection frequency from sensors can be increased at no additional cost, thus allowing higher precision and accuracy.
- Several efficiency costs can be addressed through a distributed data management, such as cleaning useless data, reducing data storage size (eliminating useless data), and etc.

Finally, we have explored and measured the effectiveness of this architecture by performing some analysis and calculation. First, with respect to the Data Acquisition block, we applied two basic data aggregation techniques, which are redundant data elimination and data compression, and compared to a real cloud based system from the smart city of Barcelona. We have shown that by applying redundant data elimination that, in some cases, the data reduction rate reaches 75%. Additionally, by applying data compression, the data reduction rate reaches an additional 78%. So the total efficiency rate, by applying both redundant data elimination and data compression, moves to almost 92%, in some cases. Note that these techniques are some of the most basic data aggregation techniques, and have been implemented to show the ease of application of such kind of optimizations in our architecture. If more sophisticated and specific techniques were used, then the performance would have been still better.

Second, we estimated the storage capacity at each F2C layer in the smart city of Barcelona, and compared it with to a cloud based system. We have shown that the size of total storage capacity (in maximum situation) will be reduced from 4,723 MB at Fog-Device (in normal data volume generation) to 1,623 MB (by applying data aggregation) and 357 MB at cloud layer (by applying both data aggregation and compression). And third, we showed that the one-way network latency is around 10 to 100ms among fog layers, and almost unpredictable (most references mentioned to 100 ms) from fog layers to cloud.

As a summary, this thesis presents numerous contributions with respect to other data management architectures analyzed so far, and extends the state of the art in the following topics:

- COmprehensive Scenario Agnostic DLC (COSA-DLC) model:This model is abstract, as it has not been designed for any specific scenario; however, it can be easily adapted to any particular scenario or science or big data environment, where data complexity has to be addressed. In fact, adapting the COSA-DLC model just requires selecting those phases that are relevant according to the specific scenario requirements
- Smart City Comprehensive Data LifeCycle (SCC-DLC) model:A data management architecture generated from a comprehensive scenario agnostic model, tailored for the particular scenario of Smart Cities.
- The distributed hierarchical F2C data management architecture provides an interesting framework for data management in the context of smart cities, according to our Smart City Comprehensive Data LifeCycle (SCC-DLC) model proposal. The Comprehensive F2C data management architecture that considers all data life cycles, including three main blocks, named the data acquisition, the data processing, and the data preservation. Data acquisition is mainly performed at fog layer 1, as well as some basic data processing and data preservation actions. The fog layer 2 can enhance the data processing and data preservations capabilities of level 1 by providing higher computing capabilities. And finally, the cloud layer is the responsible of a more complex and more sophisticated data processing over a much broader set of (presumably historical) data, as well as the responsible for permanent data preservation.
- In the data acquisition blockis mainly performed at fog layer 1to collect data from all sources and devices in the city. As long as the data are being collected, the following phases from the data acquisition block can also be performed at fog layer 1, where a reasonable amount of computing resources is available. For instance, the data filtering phase can apply filters to remove redundant data and can apply some data aggregation techniques to further reduce the amount of data to be managed. Data quality can also be implemented at this fog layer, assessing and guaranteeing higher data quality. And data description can be performed in order to tag data according to the city business model considered.
- We have shown by applying redundant data elimination that, in some cases, the data reduction rate reaches 75%. Additionally, by applying data compression, the data reduction rate increases to up to 78%. Finally, we have explored that the total efficiency rate, by applying both redundant data elimination and data compression, moves to almost 92%, in some cases. Although many other data aggregation techniques could be easily applied in this architecture, these two techniques are enough to illustrate the facility and effectiveness of such optimizations in our model.

- Data collected at fog layer 1 will be periodically moved upwards to layer 2, and data collected at layer 2 from a set of fog nodes at layer 1 will be combined and periodically moved upwards to the cloud level, which will collect the whole data set from the city. Note that data al fog layer 1 can be immediately used at this same level (real-time data), so there is not any need to move urgently these data to higher levels and, therefore, the frequency for the periodical upwards data movements can be strategically decided in order to accommodate it to the network traffic. So all this delayed updating mechanism allows optimizing many network aspects, such as network traffic and network congestion.

- In the data preservation block, data are generated at fog layer 1, but gradually moved upwards to the fog layer 2, and upwards to the cloud layer, where they will be permanently preserved. So, data generated at fog layer 1 will be temporarily stored at this level, allowing real-time applications an instant access to these data. Similarly, data gathered at fog layer 2, consisting of data received from several fog nodes at layer 1, will be temporarily stored at this level 2. This will make up a set of less recent data (as it has been received after some period of time) but from a broader area, comprising the combination of the respective fog nodes' areas at layer 1. Finally, data will be permanently preserved at cloud layer, unless any expiry time is defined.

- The different phases included in the data preservation block will be mainly executed at the cloud level, where the permanent storage is performed. Note that these phases are not urgent and, therefore, their execution can be delayed to the time in which data are received to the cloud layer. This is the case of the data classification phase, responsible for classifying and ordering data before storing, and eventually implementing the appropriate techniques for data versioning, data lineage or data provenance. And the data dissemination phase, responsible for providing a user interface for public or private access to stored data, and responsible for implementing any protection, privacy or security policies according to the city business requirements.

- In some cases, we estimated almost 70% storage reduction at layers (by applying data aggregation). Additionally, we appraised that around 91% storage reduction at layers (by applying both data aggregation and data compression).

- In the data processing block, data processing can be performed at any F2C layer, according to the requirements of the application or service. For instance, critical real-time services will be executed at fog layer 1 in order to have a faster access to the (just generated) real-time data. Alternatively, deep computing complex applications will be executed at the cloud layer.

As part of our future work we will explore more options related to all main blocks of our F2C data management (from Data Acquisition and Data Preservation to Data Processing), and continue developing other data life cycle phases of each block in our model, including extending our data

aggregation techniques, proposing some algorithms for checking data quality in F2C, practical test for applying data processing in the fog layers, real-time data analysis, proposing the algorithms for classifying data in the distributed data storage, providing a real interface for data dissemination and so on.

Publications

1- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera,"***Fog to Cloud Data Management in Smart Cities***", IEEE FTC 2017, Vancouver, Canada, November 2017.

2- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera,"***A Novel Architecture for Efficient Fog to Cloud Data Management in Smart Cities***", IEEE ICDCS 2017, Atlanta, USA, June 2017.

3- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, "***Towards a Comprehensive Data LifeCycle model for Big Data Environments***", IEEE/ACM BDCAT 2016, Shangai, China, December 2016.

4- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, X.Yin, C.Wang, "***A Data LifeCycle Model for Smart Cities***", IEEE conference on ICTC 2016, Korea, October 2016.

5- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, "***A Comprehensive Scenario Agnostic Data LifeCycle Model for an Efficient Data Complexity Management***", IEEE conference on eScience 2016, Baltimore, USA, October 2016.

6- A.Sinaeepourfard, J.Garcia, X.Masip-Bruin, E.Marín-Tordera, J.Cirera, G.Grau, F.Casaus, "***Estimating Smart City Sensors Data generation and Future Data in the City of Barcelona***", in IFIP Med-Hoc-Net 2016, Vilanova i la Geltrú, Barcelona, Spain, June 2016.

References

[1] B. Tang, Z. Chen, G. Hefferman, T. Wei, H. He, and Q. Yang, "A hierarchical distributed fog computing architecture for big data analysis in smart cities," in *The Fifth ASE International Conference on Big Data*, 2015, p. 28.

[2] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, E. Marin-Tordera, J. Cirera, G. Grau, *et al.*, "Estimating Smart City sensors data generationcurrent and future data in the city of Barcelona," in *The 15th IFIP Annual Mediterranean Ad Hoc Networking Workshop*, 2016, in press, (preprint version athttp://craax.upc.edu/images/Publications/conferences/2016/medhonet_16_amir.pdf).

[3] J. Wang, Y. Tang, M. Nguyen, and I. Altintas, "A Scalable Data Science Workflow Approach for Big Data Bayesian Network Learning," in *Proceedings of the 2014 IEEE/ACM International Symposium on Big Data Computing (BDC)*, 2014, pp. 16-25.

[4] V. Kolias, I. Anagnostopoulos, and E. Kayafas, "A Covering Classification Rule Induction Approach for Big Datasets," in *Proceedings of the 2014 IEEE/ACM International Symposium on Big Data Computing*, 2014, pp. 45-53.

[5] H. Hu, Y. Wen, T.-S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial," *Journals & Magazines on IEEE Access*, vol. 2, pp. 652-687, 2014.

[6] Y. Demchenko, Z. Zhao, P. Grosso, A. Wibisono, and C. De Laat, "Addressing big data challenges for scientific data infrastructure," in *IEEE 4th International Conference on Cloud Computing Technology and Science (CloudCom)*, 2012, pp. 614-617.

[7] R. Grunzke, A. Aguilera, W. E. Nagel, S. Herres-Pawlis, A. Hoffmann, J. Krüger, *et al.*, "Managing complexity in distributed Data Life Cycles enhancing scientific discovery," in *IEEE 11th International Conference on E-Science (e-Science)*, 2015, pp. 371-380.

[8] S. Henry, S. Hoon, M. Hwang, D. Lee, and M. D. DeVore, "Engineering trade study: extract, transform, load tools for data migration," in *IEEE Conference on Design Symposium, Systems and Information Engineering*, 2005, pp. 1-8.

[9] S. Kurunji, T. Ge, B. Liu, and C. X. Chen, "Communication cost optimization for cloud Data Warehouse queries," in *IEEE 4th International Conference on Cloud Computing Technology and Science (CloudCom)*, 2012, pp. 512-519.

[10] F. L. F. Almeida and C. Calistru, "The main challenges and issues of big data management," *International Journal of Research Studies in Computing*, vol. 2, 2012.

[11] A. Levitin and T. Redman, "A model of the data (life) cycles with application to quality," *Journal of Information and Software Technology on Elsevier*, vol. 35, pp. 217-223, 1993.

[12] W. K. Michener and M. B. Jones, "Ecoinformatics: supporting ecology as a data-intensive science," *Journal of Trends in ecology & evolution*, vol. 27, pp. 85-93, 2012.

[13] J. Rüegg, C. Gries, B. Bond-Lamberty, G. J. Bowen, B. S. Felzer, N. E. McIntyre, *et al.*, "Completing the Data Life Cycle: using information management in macrosystems ecology research," *Journal of Frontiers in Ecology and the Environment*, vol. 12, pp. 24-30, 2014.

[14] R. Y. Wang and D. M. Strong, "Beyond accuracy: What data quality means to data consumers," *Journal of management information systems*, pp. 5-33, 1996.

[15] J. L. Faundeen, T. E. Burley, J. A. Carlino, D. L. Govoni, H. S. Henkel, S. L. Holl, *et al.*, "The United States Geological Survey Science Data Lifecycle Model: U.S. Geological Survey Open-File Report 2013–1265, 4 p.," US Geological Survey 2331-1258, 2013.

[16] X.Masip, E.Marín, A.Jukan, G.J.Ren, and G.Tashakor, "Foggy clouds and cloudy fogs: a real need for coordinated management of fog-to-cloud (F2C) computing systems," *Jounal of IEEE Wireless Communications Magazine*, 2016, in press, (preprint version at http://www.craax.upc.edu/images/Publications/journals/Fog-to-cloud_preprint.pdf).

[17] H. Jagadish, J. Gehrke, A. Labrinidis, Y. Papakonstantinou, J. M. Patel, R. Ramakrishnan, *et al.*, "Big data and its technical challenges," *Communications of the ACM*, vol. 57, pp. 86-94, 2014.

[18] S. D. Kahn, "On the future of genomic data," *Science(Washington)*, vol. 331, pp. 728-729, 2011.

[19] O. p. d. o. t. g. o. C. O. D. gencat). (2015). *Data collected by the app*. Available: http://dadesobertes.gencat.cat/en/cercador/detall-cataleg/?id=7710

[20] G. d. Catalunya. *Cobertura mòbil Application*. Available: http://coberturamobil.gencat.cat/web/index_en

[21] A. Paul, "IoT and Big Data towards a Smart City," *Journal of World Scientific News*, vol. 41, p. 54, 2016.

[22] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the Internet of Things using Big Data analytics," *Computer Networks*, vol. 101, pp. 63-80, 6/4/ 2016.

[23] F. Mehdipour, B. Javadi, and A. Mahanti, "FOG-Engine: Towards Big Data Analytics in the Fog," in *Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 2016 IEEE 14th Intl C*, 2016, pp. 640-646.

[24] J. Lassinantti, "Public Sector Open Data."

[25] USGS. *USGS Data Lifecycle Overview*. Available: Available on: http://www.usgs.gov/datamanagement/why-dm/lifecycleoverview.php

[26] Z. Zhi-Hua, N. V. Chawla, J. Yaochu, and G. J. Williams, "Big Data Opportunities and Challenges: Discussions from Data Analytics Perspectives [Discussion Forum]," *Computational Intelligence Magazine, IEEE*, vol. 9, pp. 62-74, 2014.

[27] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani, and S. U. Khan, "The rise of "big data" on cloud computing: Review and open research issues," *Journal of Information Systems on Elsevier*, vol. 47, pp. 98-115, 2015.

[28] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," *Journal on Mobile Networks and Applications*, vol. 19, pp. 171-209, 2014.

[29] M. Cox and D. Ellsworth, "Managing big data for scientific visualization," in *ACM Siggraph*, 1997, p. 21.

[30] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, *et al.*, "Big data: The next frontier for innovation, competition, and productivity," 2011.

[31] R. L. Villars, C. W. Olofson, and M. Eastwood, "Big data: What it is and why you should care," *White Paper, IDC*, 2011.

[32] D. Gordijenko, "Big Security for Big Data: Addressing Security Challenges for the Big Data Infrastructure," in *Secure Data Management: 10th VLDB Workshop, SDM 2013, Trento, Italy, August 30, 2013, Proceedings*, 2014, p. 76.

[33] S. J. Samuel, R. Koundinya, K. Sashidhar, and C. Bharathi, "A SURVEY ON BIG DATA AND ITS RESEARCH CHALLENGES," 2006.

[34] E. M. Micheni, "Diffusion of Big Data and Analytics in Developing Countries," 2015.

[35] J. Bloomberg, "The Big Data Long Tail," ed, 2013.

[36] P. Zikopoulos, K. Parasuraman, T. Deutsch, J. Giles, and D. Corrigan, *Harness the Power of Big Data The IBM Big Data Platform*: McGraw Hill Professional, 2012.

[37] J. J. Berman, *Principles of big data: preparing, sharing, and analyzing complex information*: Newnes, 2013.

[38] D. Laney, "3D data management: Controlling data volume, velocity and variety," *META Group Research Note*, vol. 6, p. 70, 2001.

[39] D. E. O'Leary, "Artificial intelligence and big data," *Journal of IEEE Intelligent Systems*, vol. 28, pp. 0096-99, 2013.

[40] I. Gartner. *Big Data*. Available: http://www.gartner.com/it-glossary/big-data

[41] R. Rossi and K. Hirama, "Characterizing Big Data Management," *Journal on Issues in Informing Science and Information Technology (IISIT)*, vol. 12, 2015.

[42] R. Narasimhan and T. Bhuvaneshwari, "Big Data–A Brief Study."

[43] E. MCNULTY. (2014). *Understanding Big Data: The Seven V's*. Available: http://dataconomy.com/seven-vs-big-data/

[44] (2015). *Understanding the 7 V's of Big Data*. Available: http://www.optimusinfo.com/blog/understanding-the-7-vs-of-big-data/

[45] J. GURIN, "DRIVING INNOVATION WITH OPEN DATA," *THE FUTURE OF*, p. 55, 2014.

[46] J. Ridgway and A. Smith, "Open data, official statistics and statistics education: threats, and opportunities for collaboration," in *Proceedings of the Joint IASEIAOS Satellite Conference "Statistics Education for Progress", Macao, China*, 2013.

[47] G.-J. Ren and S. Glissmann, "Identifying information assets for open data: the role of business architecture and information quality," in *Commerce and Enterprise Computing (CEC), 2012 IEEE 14th International Conference on*, 2012, pp. 94-100.

[48] X. Masip-Bruin, G.-J. Ren, R. Serral-Gracià, and M. Yannuzzi, "Unlocking the Value of Open Data with a Process-Based Information Platform," in *Business Informatics (CBI), 2013 IEEE 15th Conference on*, 2013, pp. 331-337.

[49] J. Gurin, *Open data now: the secret to hot startups, smart investing, savvy marketing, and fast innovation*: McGraw Hill Education, 2014.
[50] M. S. Fox, "City data: Big, open and linked," *Department of Mechanical and Industrial Engineering University of Toronto*, 2013.
[51] A. M. Al-Khouri, "Open Data: A Paradigm Shift in the Heart of Government," *Journal of Public Administration and Governance*, vol. 4, pp. Pages 217-244, 2014.
[52] T. Jetzek, M. Avital, and N. Bjørn-Andersen, "The Value of Open Government Data," *Geoforum Perspektiv*, vol. 23, pp. 48-57, 2013.
[53] J. Gurin, "Open Governments, Open Data: A New Lever for Transparency, Citizen Engagement, and Economic Growth," *SAIS Review of International Affairs*, vol. 34, pp. 71-82, 2014.
[54] B. Ubaldi, "Open Government Data," 2013.
[55] J. Sheridan and J. Tennison, "Linking UK Government Data," in *LDOW*, 2010.
[56] (2015). *Open Government Data*. Available: http://www.data.gov/opendatasites
[57] A. Zuiderwijk and M. Janssen, "The negative effects of open government data-investigating the dark side of open data," in *Proceedings of the 15th Annual International Conference on Digital Government Research*, 2014, pp. 147-152.
[58] C. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on Big Data," *Information Sciences*, vol. 275, pp. 314-347, 2014.
[59] N. B. D. PWG, "Draft NIST Big Data Interoperability Framework: Volume 4, Security and Privacy," 2015.
[60] P. Struijs, B. Braaksma, and P. J. Daas, "Official statistics and Big Data," *Big Data & Society*, vol. 1, p. 2053951714538417, 2014.
[61] E. Sahafizadeh and M. A. Nematbakhsh, "A Survey on Security Issues in Big Data and NoSQL," 2015.
[62] (2015). *Dimensions of Big Data*. Available: http://www.klarity-analytics.com/392-dimensions-of-big-data.html
[63] K. Normandeau. (2013). *Beyond Volume, Variety and Velocity is the Issue of Big Data Veracity*. Available: http://insidebigdata.com/2013/09/12/beyond-volume-variety-velocity-issue-big-data-veracity/
[64] K. C. Desouza and K. L. Smith. Big Data for Social Innovation [Online]. Available: http://ssir.org/articles/entry/big_data_for_social_innovation
[65] J. Hurwitz, A. Nugent, F. Halper, and M. Kaufman. *How to Ensure the Validity, Veracity, and Volatility of Big Data*. Available: http://www.dummies.com/how-to/content/how-to-ensure-the-validity-veracity-and-volatility.html
[66] M. Ali-ud-din Khan, M. F. Uddin, and N. Gupta, "Seven V's of Big Data understanding Big Data to extract value," in *American Society for Engineering Education (ASEE Zone 1), 2014 Zone 1 Conference of the*, 2014, pp. 1-5.
[67] B. D. Alliance. (2015). *What is Big Data?* Available: http://www.bigdata-alliance.org/what-is-big-data/
[68] B. Vorhies. (2013). *How Many "V"s in Big Data – The Characteristics that Define Big Data*. Available: http://data-magnum.com/how-many-vs-in-big-data-the-characteristics-that-define-big-data/
[69] T. Shan. (2014). *Big Data Characterized*. Available: http://cloudonomic.blogspot.com.es/2014/11/big-data-characterized.html
[70] J. A. di Paolantonio and B. B. Goewey. (2012). *Big Data, What is it Exactly? Datamensional's Take*. Available: http://www.datamensional.com/big-data/
[71] W. Ray. (2012). *Beyond The Three V's of Big Data – Viscosity and Virality*. Available: http://blog.softwareinsider.org/2012/02/27/mondays-musings-beyond-the-three-vs-of-big-data-viscosity-and-virality/
[72] K. Desouza, "Realizing the Promise of Big Data," ed: Washington, DC: IBM Center for the Business of Government, 2014.
[73] K. Krishnan, *Data warehousing in the age of big data*: Newnes, 2013.
[74] H. Li and X. Lu, "Challenges and Trends of Big Data Analytics," in *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2014 Ninth International Conference on*, 2014, pp. 566-567.
[75] L. Amodeo. (2014). *BIG DATA 6V: VOLUME, VARIETY, VELOCITY, VARIABILITY, VERACITY, COMPLEXITY*. Available: https://wydata.wordpress.com/2014/12/24/big-data-volume-variety-velocity-variability-veracity-complexity/
[76] M. Van Rijmenam. *Why the 3v's are not sufficient to describe big data*. Available: https://datafloq.com/read/3vs-sufficient-describe-big-data/166
[77] M. Basu and T. K. Ho, *Data complexity in pattern recognition*: Springer Science & Business Media, 2006.

[78] A. Farhangfar, L. Kurgan, and J. Dy, "Impact of imputation of missing values on classification error for discrete data," *Pattern Recognition*, vol. 41, pp. 3692-3705, 12// 2008.

[79] X. Wu and X. Zhu, "Mining with noise knowledge: error-aware data mining," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 38, pp. 917-932, 2008.

[80] N. V. Chawla, "Data mining for imbalanced datasets: An overview," in *Data mining and knowledge discovery handbook*, ed: Springer, 2005, pp. 853-867.

[81] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera, "A unifying view on dataset shift in classification," *Pattern Recognition*, vol. 45, pp. 521-530, 2012.

[82] IBM. *The Four V's of Big Data*. Available: http://www.ibmbigdatahub.com/infographic/four-vs-big-data

[83] I. Zahumenský and J. SHMI, "Guidelines on quality control procedures for data from automatic weather stations," *World Meteorological Organization, Switzerland*, 2004.

[84] DataONE. *Tutorials on Data Management*. Available: https://www.dataone.org/sites/all/documents/L05_Exercise.pdf

[85] B. Arthur. *The Difference Between Quality Assurance and Quality Control*. Available: http://www.dialog.com.au/open-dialog/the-difference-between-quality-assurance-and-quality-control/

[86] *Quality Assurance vs. Quality Control*. Available: http://www.diffen.com/difference/Quality_Assurance_vs_Quality_Control

[87] L. Wang, G. Wang, and C. A. Alexander, "Big Data and Visualization: Methods, Challenges and Technology Progress," *Digital Technologies*, vol. 1, pp. 33-38, 2015.

[88] H. Hu, Y. Wen, T.-S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial," *Magazines on IEEE Access*, vol. 2, pp. 652-687, 2014.

[89] K. L. Hanson, T. A. Bakker, M. A. Svirsky, A. C. Neuman, and N. Rambo, "Informationist Role: Clinical Data Management in Auditory Research," *Journal of eScience Librarianship*, vol. 2, p. 7, 2013.

[90] J. M. Schopf, "Treating data like software: a case for production quality data," in *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries*, 2012, pp. 153-156.

[91] W. Lenhardt, S. Ahalt, B. Blanton, L. Christopherson, and R. Idaszak, "Data management Lifecycle and Software Lifecycle management in the context of conducting science," *Journal of Open Research Software*, vol. 2, 2014.

[92] M. Emaldi, O. Peña, J. Lázaro, and D. López-de-Ipiña, "Linked Open Data as the Fuel for Smarter Cities," in *Modeling and Processing for Next-Generation Big-Data Technologies*, ed: Springer, 2015, pp. 443-472.

[93] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a Smart City through Internet of Things," *Journal of Internet of Things Journal on IEEE*, vol. 1, pp. 112-121, 2014.

[94] A. Burton and A. Treloar, "Publish my data: a composition of services from ANDS and ARCS," in *IEEE 5th International Conference on E-Science (e-Science)*, 2009, pp. 164-170.

[95] A. Sinaeepourfard, X. Masip-Bruin, J. Garcia, and E. Marín-Tordera, "A Survey on Data Lifecycle Models: Discussions toward the 6Vs Challenges," Technical Report (UPC-DAC-RR-ANA-2015-1), 2015.

[96] S. Kethers, X. Shen, A. E. Treloar, and R. G. Wilkinson, "Discovering Australia's research data," in *Proceedings of the 10th annual joint conference on Digital libraries*, 2010, pp. 345-348.

[97] A. Ball, "Review of data management lifecycle models," University of Bath2012.

[98] A. Burton and A. Treloar, "Designing for discovery and re-use: the 'ANDS data sharing verbs' approach to service decomposition," *International Journal of Digital Curation*, vol. 4, pp. 44-56, 2009.

[99] ANDS. *Australian National Data Service (ANDS)*. Available: Established in 2008, Available on: http://www.ands.org.au/

[100] BLM. *The Bureau of Land Management: Who We Are, What We Do*. Available: http://www.blm.gov/wo/st/en/info/About_BLM.html

[101] CEOS, "CEOS Data Life Cycle Models and Concepts Version 1.0," 2011.

[102] A. A. Atayero and O. Feyisetan, "Security issues in cloud computing: The potentials of homomorphic encryption," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 2, pp. 546-552, 2011.

[103] W. K. Michener, S. Allard, A. Budden, R. B. Cook, K. Douglass, M. Frame, *et al.*, "Participatory design of DataONE—Enabling cyberinfrastructure for the biological and environmental sciences," *Elsevier journal in Ecological Informatics*, vol. 11, pp. 5-15, 9// 2012.

[104] C. Eaker, A. K. Thomer, E. Johns, and K. Siddell, "How information science professionals add value in a scientific research center," in *iConference 2013 Proceedings*, 2013.

[105] S. Higgins, "The DCC curation lifecycle model," *International Journal of Digital Curation*, vol. 3, pp. 134-140, 2008.

[106] P. Constantopoulos, C. Dallas, I. Androutsopoulos, S. Angelis, A. Deligiannakis, D. Gavrilis, *et al.*, "DCC&U: An extended digital curation lifecycle model," *International Journal of Digital Curation*, vol. 4, pp. 34-45, 2009.

[107] S. Higgins, "DCC DIFFUSE standards frameworks: a standards path through the curation lifecycle," *International Journal of Digital Curation*, vol. 4, pp. 60-67, 2009.

[108] A. Whyte, D. Job, S. Giles, and S. Lawrie, "Meeting curation challenges in a neuroimaging group," *International Journal of Digital Curation*, vol. 3, pp. 171-181, 2008.

[109] G. Goth, "Preserving digital data," *Magazine on Communications of the ACM (Commun. ACM)*, vol. 55, pp. 11-13, 2012.

[110] CEOS, "CEOS Data Life Cycle Models and Concepts Version 1.2," 2012.

[111] DIGITALNZ. *Helping to make New Zealand digital content easy to find, share, and use*. Available: Established in 2008, Available on: http://www.digitalnz.org/about

[112] FGDC, "Stages of the Geospatial Data Lifecycle pursuant to OMB Circular A–16, sections 8(e)(d), 8(e)(f), and 8(e)(g) " 2010.

[113] S. Auer, L. Bühmann, C. Dirschl, O. Erling, M. Hausenblas, R. Isele, *et al.*, "Managing the life-cycle of linked data with the LOD2 stack," in *The Semantic Web–ISWC 2012*, ed: Springer, 2012, pp. 1-16.

[114] S. Auer, J. Lehmann, A.-C. N. Ngomo, and A. Zaveri, "Introduction to linked data and its lifecycle on the web," in *Reasoning Web. Semantic Technologies for Intelligent Data Access*, ed: Springer, 2013, pp. 1-90.

[115] M. S. University. (2011). *Records Management*. Available: Available on: http://archives.msu.edu/records/

[116] S. Hodson, "Meeting the Research Data Challenge," 2011.

[117] X. Yu and Q. Wen, "A view about cloud data security from data life cycle," in *IEEE International Conference on Computational Intelligence and Software Engineering (CiSE)*, 2010, pp. 1-4.

[118] C. Kyriazopoulou, "Smart City technologies and architectures: A literature review," in *Smart Cities and Green ICT Systems (SMARTGREENS), 2015 International Conference on*, 2015, pp. 1-12.

[119] C. Kyriazopoulou, "Smart City technologies and architectures: A literature review," in *International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*, 2015, pp. 1-12.

[120] (2015). *Seven Spanish cities among the Top 50 smartest cities in Europe*. Available: http://marcaespana.es/en/news/society/seven-spanish-cities-among-top-50-smartest-cities-europe

[121] I. B. School. (2015). *IESE Cities in Motion Index*. Available: http://www.iese.edu/research/pdfs/ST-0366-E.pdf

[122] I. A. T. Hashem, V. Chang, N. B. Anuar, K. Adewole, I. Yaqoob, A. Gani, *et al.*, "The role of big data in Smart City," *International Journal of Information Management*, vol. 36, pp. 748-758, 2016.

[123] M. Ma, P. Wang, and C.-H. Chu, "Data management for internet of things: challenges, approaches and opportunities," in *Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing*, 2013, pp. 1144-1151.

[124] A. Gaur, B. Scotney, G. Parr, and S. McClean, "Smart City Architecture and its Applications Based on IoT," *Procedia Computer Science*, vol. 52, pp. 1089-1094, // 2015.

[125] T. Fan and Y. Chen, "A scheme of data management in the Internet of Things," in *Network Infrastructure and Digital Content, 2010 2nd IEEE International Conference on*, 2010, pp. 110-114.

[126] D. J. Abadi, "Data management in the cloud: Limitations and opportunities," *IEEE Data Eng. Bull.*, vol. 32, pp. 3-12, 2009.

[127] M. Firdhous, O. Ghazali, and S. Hassan, "Fog computing: Will it be the future of cloud computing?," 2014.

[128] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the suitability of fog computing in the context of internet of things," *IEEE Transactions on Cloud Computing*, 2015.

[129] V. Souza, W. Ramírez, X. Masip-Bruin, E. Marín-Tordera, G. Ren, and G. Tashakor, "Handling service allocation in combined fog-cloud scenarios," in *IEEE International Conference on Communications (ICC)*, 2016, pp. 1-5.

[130] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, 2012, pp. 13-16.

[131] A. V. Dastjerdi, H. Gupta, R. N. Calheiros, S. K. Ghosh, and R. Buyya, "Fog computing: Principles, architectures, and applications," *arXiv preprint arXiv:1601.02752*, 2016.

[132] V. B. Souza, X. Masip-Bruin, E. Marin-Tordera, W. Ramírez, and S. Sanchez, "Towards Distributed Service Allocation in Fog-to-Cloud (F2C) Scenarios," in *IEEE Global Communications Conference (GLOBECOM)*, 2016, pp. 1-6.

[133] X. Masip-Bruin, E. Marín-Tordera, A. Gómez, V. Barbosa, and A. Alonso, "Will it be cloud or will it be fog? F2C, A novel flagship computing paradigm for highly demanding services," in *Future Technologies Conference (FTC)*, 2016, pp. 1129-1136.

[134] I. Stojmenovic, S. Wen, X. Huang, and H. Luan, "An overview of fog computing and its security issues," *Concurrency and Computation: Practice and Experience*, vol. 28, pp. 2991-3005, 2016.

[135] X. Ouyang, D. Irwin, and P. Shenoy, "SpotLight: An Information Service for the Cloud," in *36th IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2016, pp. 425-436.

[136] X. Hu, A. Ludwig, A. Richa, and S. Schmid, "Competitive Strategies for Online Cloud Resource Allocation with Discounts: The 2-Dimensional Parking Permit Problem," in *35th IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2015, pp. 93-102.

[137] S. Kannan, A. Gavrilovska, and K. Schwan, "Cloud4Home--Enhancing Data Services with@ Home Clouds," in *31st IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2011, pp. 539-548.

[138] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, E. Marin-Tordera, X. Yin, and C. Wang, "A data lifeCycle model for smart cities," in *IEEE International Conference on ICT Convergence (ICTC)*, 2016, pp. 400-405.

[139] T. V. N. Rao, A. Khan, M. Maschendra, and M. K. Kumar, "A Paradigm Shift from Cloud to Fog Computing," *International Journal of Science, Engineering and Computer Technology*, vol. 5, p. 385, 2015.

[140] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a Smart City through internet of things," *IEEE Internet of Things Journal*, vol. 1, pp. 112-121, 2014.

[141] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Journal of Future Generation Computer Systems on Elsevier*, vol. 29, pp. 1645-1660, 2013.

[142] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a Smart City through internet of things," *Journal of Internet of Things on IEEE*, vol. 1, pp. 112-121, 2014.

[143] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the Internet of Things using Big Data analytics," *Journal of Computer Networks on Elsevier*, vol. 101, pp. 63-80, 2016.

[144] P. A. Pena, D. Sarkar, and P. Maheshwari, "A Big-Data Centric Framework for Smart Systems in the World of Internet of Everything," in *The 2015 International Conference on Computational Science and Computational Intelligence (CSCI'15)*, 2015, pp. 306-311.

[145] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the Suitability of Fog Computing in the Context of Internet of Things," *Journal of IEEE Transactions on Cloud Computing*, 2015.

[146] W. P. Review. (2016). *Barcelona Population 2016*. Available: http://worldpopulationreview.com/world-cities/barcelona-population/

[147] D. d'Estadistica. *Barcelona statistics, districts and neighborhoods*. Available: http://www.bcn.cat/estadistica/angles/index.htm

[148] A. d. Barcelona. *City OS Architecture*. Available: http://ibarcelona.bcn.cat/sites/default/files/city_os_-_inside.pdf

[149] F. Mehdipour, B. Javadi, and A. Mahanti, "FOG-Engine: Towards Big Data Analytics in the Fog," in *IEEE International Conference on Big Data Intelligence and Computing*, 2016, pp. 640-646.

[150] Libelium. (2014). *Libelium Links with Barcelona Smart Cities Cloud Platform*. Available: http://www.libelium.com/libelium-links-with-barcelona-smart-cities-cloud-platform/#!prettyPhoto

[151] A. d. Barcelona. *Platform of Sensors and Actuators of Barcelona*. Available: http://ibarcelona.bcn.cat/en/smart-cities/platform-sensors-and-actuators-barcelona

[152] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a Smart City through internet of things," *IEEE Journal on Internet of Things* vol. 1, pp. 112-121, 2014.

[153] A. Sinaeepourfard, J. Garcia, X. Masip, E. Marin-Tordera, J. Cirera, G. Grau, *et al.*, "Estimating Smart City sensors data generation current and future data in the city of Barcelona," in *The 15th IFIP Annual Mediterranean Ad Hoc Networking Workshop*, 2016, in press.

[154] R. Gómez-Enrich, M. López-Vivancos, M. Mestre-Vidal, J. Prats-Prat, and A. Rovira-Fernández, "Towards the integral management of library collections at the Technical University of Catalonia (UPC)," presented at the 25th IATUL Annual Conference on The International Association of Scientific and Technological University Libraries (IATUL), Krakow, Poland, 2004.

[155] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, and E. Marín-Torder, "Towards a comprehensive data lifecycle model for big data environments," in *The 3rd IEEE/ACM International Conference on Big Data Computing, Applications and Technologies 2016 (BDCAT)*, 2016, pp. 100-106.

[156] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, and E. Marín-Tordera, "A comprehensive scenario agnostic Data LifeCycle model for an efficient data complexity management," in *IEEE 12th International Conference on e-Science (e-Science)*, 2016, pp. 276-281.

[157] T. Fan and Y. Chen, "A scheme of data management in the Internet of Things," in *IEEE International Conference on Network Infrastructure and Digital Content*, 2010, pp. 110-114.

[158] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," *Journal of Mobile Networks and Applications*, vol. 19, pp. 171-209, 2014.

[159] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the Internet of Things using Big Data analytics," *Elsevier Jounral on Computer Networks*, vol. 101, pp. 63-80, 6/4/ 2016.

[160] Z. Khan, S. L. Kiani, and K. Soomro, "A framework for cloud-based context-aware information services for citizens in smart cities," *Journal of Cloud Computing*, vol. 3, p. 14, 2014.

[161] M. Abu-Elkheir, M. Hayajneh, and N. A. Ali, "Data management for the internet of things: Design primitives and solution," *Journal of Sensors*, vol. 13, pp. 15582-15612, 2013.

[162] A. Botta, W. De Donato, V. Persico, and A. Pescapé, "On the integration of cloud computing and internet of things," in *IEEE International Conference on Future Internet of Things and Cloud (FiCloud)*, 2014, pp. 23-30.

[163] H. Hu, Y. Wen, T.-S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial," *IEEE Access*, vol. 2, pp. 652-687, 2014.

[164] D. Emmanouil and D. Nikolaos, "Big data analytics in prevention, preparedness, response and recovery in crisis and disaster management," in *The 18th International Conference on Circuits, Systems, Communications and Computers (CSCC 2015), Recent Advances in Computer Engineering Series*, 2015, pp. 476-482.

[165] I. A. T. Hashem, V. Chang, N. B. Anuar, K. Adewole, I. Yaqoob, A. Gani, *et al.*, "The role of big data in Smart City," *International Journal on Information Management*, vol. 36, pp. 748-758, 10// 2016.

[166] R. Wenge, X. Zhang, C. Dave, L. Chao, and S. Hao, "Smart City architecture: A technology guide for implementation and design challenges," *IEEE Journal on China Communications*, vol. 11, pp. 56-69, 2014.

[167] A. Papageorgiou, M. Schmidt, J. Song, and N. Kami, "Smart m2m data filtering using domain-specific thresholds in domain-agnostic platforms," in *IEEE International Congress on big data (BigData congress)* 2013, pp. 286-293.

[168] P. Patil and U. Kulkarni, "Delay Efficient Distributed Data Aggregation Algorithm in Wireless Sensor Networks," *International Journal of Computer Applications*, vol. 69, 2013.

[169] N. Karthick and X. A. Kalrani, "A Survey on Data Aggregation in Big Data and Cloud Computing," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 17, pp. 28-32, 2014.

[170] J.-Y. Chen, G. Pandurangan, and D. Xu, "Robust computation of aggregates in wireless sensor networks: distributed randomized algorithms and analysis," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, pp. 987-1000, 2006.

[171] S. Sirsikar and S. Anavatti, "Issues of Data Aggregation Methods in Wireless Sensor Network: A Survey," *Journal of Procedia Computer Science on Elsevier*, vol. 49, pp. 194-201, 2015.

[172] T. He, L. Gu, L. Luo, T. Yan, J. A. Stankovic, and S. H. Son, "An overview of data aggregation architecture for real-time tracking with sensor networks," in *20th IEEE International Parallel & Distributed Processing Symposium*, 2006, p. 8 pp.

[173] P. Jesus, C. Baquero, and P. S. Almeida, "A survey of distributed data aggregation algorithms," *IEEE Communications Surveys & Tutorials*, vol. 17, pp. 381-404, 2015.

[174] T. Knap and J. Michelfeit, "Linked Data Aggregation Algorithm: Increasing Completeness and Consistency of Data," *Journal provided by Charles University*, 2012.

[175] T. Knap, J. Michelfeit, and M. Necasky, "Linked Open Data Aggregation: Conflict Resolution and Aggregate Quality," presented at the Proceedings of the 2012 IEEE 36th Annual Computer Software and Applications Conference Workshops, 2012.

[176] S. Chhabra and D. Singh, "Data Fusion and Data Aggregation/Summarization Techniques in WSNs: A Review," *International Journal of Computer Applications*, vol. 121, 2015.

[177] H. R. Dhasian and P. Balasubramanian, "Survey of data aggregation techniques using soft computing in wireless sensor networks," *Journal of IET Information Security*, vol. 7, pp. 336-342, 2013.

[178] A. V. Levitin and T. C. Redman, "A model of the data (life) cycles with application to quality," *Elsevier Journal on Information and Software Technology*, vol. 35, pp. 217-223, 1993.

[179] A. Klein and W. Lehner, "Representing data quality in sensor data streaming environments," *Journal of Data and Information Quality (JDIQ)*, vol. 1, p. 10, 2009.

[180] A. Karkouch, H. Mousannif, H. Al Moatassime, and T. Noel, "Data quality in internet of things: A state-of-the-art survey," *Elsevier Journal on Network and Computer Applications*, vol. 73, pp. 57-81, 2016.

[181] D. M. Strong, Y. W. Lee, and R. Y. Wang, "Data quality in context," *Communications of the ACM*, vol. 40, pp. 103-110, 1997.

[182] Y. Wand and R. Y. Wang, "Anchoring data quality dimensions in ontological foundations," *Communications of the ACM*, vol. 39, pp. 86-95, 1996.

[183] Y. W. Lee, D. M. Strong, B. K. Kahn, and R. Y. Wang, "AIMQ: a methodology for information quality assessment," *Information & Management*, vol. 40, pp. 133-146, 12// 2002.

[184] S. R. Jeffery, G. Alonso, M. J. Franklin, W. Hong, and J. Widom, "Declarative support for sensor data cleaning," in *International Conference on Pervasive Computing*, 2006, pp. 83-100.

[185] Z. Khan, D. Ludlow, R. McClatchey, and A. Anjum, "An architecture for integrated intelligence in urban management using cloud computing," *Journal of Cloud Computing: Advances, Systems and Applications*, vol. 1, p. 1, 2012.

[186] P. Barnaghi, M. Bermudez-Edo, and R. Tönjes, "Challenges for quality of data in smart cities," *Journal of Data and Information Quality (JDIQ)*, vol. 6, p. 6, 2015.

[187] T. Plagemann, V. Goebel, A. Mauthe, L. Mathy, T. Turletti, and G. Urvoy-Keller, "From content distribution networks to content networks — issues and challenges," *Elsevier Jounral on Computer Communications*, vol. 29, pp. 551-562, 2006/03/06/.

[188] G. Płoszajski, "Metadata in Long-Term Digital Preservation," in *Digital Preservation: Putting It to Work*, ed: Springer, 2017, pp. 15-61.

[189] H. Jeung, S. Sarni, I. Paparrizos, S. Sathe, K. Aberer, N. Dawes, *et al.*, "Effective metadata management in federated sensor networks," in *IEEE International Conference onSensor Networks, Ubiquitous, and Trustworthy Computing (SUTC)*, 2010, pp. 107-114.

[190] X. Liu, A. Heller, and P. S. Nielsen, "CITIESData: a Smart City data management framework," *Journal of Knowledge and Information Systems*, pp. 1-24, 2017.

[191] B. Confais, A. Lèbre, and B. Parrein, "An Object Store Service for a Fog/Edge Computing Infrastructure based on IPFS and Scale-out NAS," in *1st IEEE International Conference on Fog and Edge Computing (ICFEC'2017)*, 2017.

[192] A. d. Barcelona. (2009). *Superficie de las divisiones administrativas*. Available: http://www.bcn.cat/estadistica/angles/dades/timm/tterr/a2009/S04.htm

[193] I. d. E. d. Catalunya. *idescat (Official Statistics website of Catalonia)*. Available: http://www.idescat.cat/en/

[194] A. Ahmad, M. M. Rathore, A. Paul, and S. Rho, "Defining Human Behaviors Using Big Data Analytics in Social Internet of Things," in *Advanced Information Networking and Applications (AINA), 2016 IEEE 30th International Conference on*, 2016, pp. 1101-1107.

[195] *Water Meters*. Available: http://data.surrey.ca/dataset/water-meters

[196] A. d. Barcelona. (2016). *Ocupación media de los domicilios*. Available: http://www.bcn.cat/estadistica/angles/dades/tpob/llars/padro/a2016/persones/ocu01.htm

[197] A. d. Barcelona. (2016). *Evolución del total de la población. 1970-2016*. Available: http://www.bcn.cat/estadistica/angles/dades/tpob/pad/ine/evo/t0101.htm

[198] S. Uppoor and M. Fiore, "Large-scale urban vehicular mobility for networking research," in *Vehicular Networking Conference (VNC), 2011 IEEE*, 2011, pp. 62-69.

[199] D. Naboulsi and M. Fiore, "On the instantaneous topology of a large-scale urban vehicular network: the cologne case," in *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing*, 2013, pp. 167-176.

[200] A. d. Barcelona. (2016). *Evolución del parque de vehículos*. Available: http://www.bcn.cat/estadistica/angles/dades/economia/vehicles/vehievo/t15.htm

[201] O. D. gencat. *Data collected by the app*. Available: http://dadesobertes.gencat.cat/en/cercador/detall-cataleg/?id=7710

[202] S. Dey, A. Chakraborty, S. Naskar, and P. Misra, "Smart City surveillance: Leveraging benefits of cloud data stores," in *Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on*, 2012, pp. 868-876.

[203] Aitor. (2012). *The timing of the traffic lights*. Available: http://thingsfrombarcelona.blogspot.com.es/2012/02/timing-of-traffic-lights.html

[204] B. Airport. (2017). *BARCELONA AIRPORT - EL PRAT*. Available: http://www.barcelona-airport.com/eng/terminal_T1_T2.php

[205] PKWARE. *APPNOTE*. Available: https://support.pkware.com/display/PKZIP/APPNOTE.

[206] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, and E. Marín-Tordera, "Fog-to-Cloud (F2C) data management for Smart Cities " presented at the Future Technologies Conference (FTC), Vancouver, BC, Canada, 2017, "in press".

[207] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, and E. Marín-Tordera, "A Novel Architecture for Efficient Fog to Cloud Data Management in Smart Cities," presented at the The 37th IEEE International Conference on Distributed Computing Systems (ICDCS 2017), Atlanta, GA, USA, 2017, "in press".

[208] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, E. Marin-Tordera, X. Yin, and C. Wang, "A data lifeCycle model for smart cities," in *International Conference on Information and Communication Technology Convergence (ICTC)*, 2016, pp. 400-405.

[209] A. Sinaeepourfard, J. Garcia, X. Masip-Bruin, E. Marín-Tordera, J. Cirera, G. Grau, *et al.*, "Estimating Smart City sensors data generation," in *Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, Vilanova i La Geltru, Barcelona, Spain, 2016, pp. 1-8.

[210] T. Traczyk, W. Ogryczak, P. Pałka, and T. Śliwiński, "Digital Preservation: Putting It to Work," ed: Springer, 2017.

[211] X. Zhao, Z. Li, and L. Zeng, "A hierarchical storage strategy based on block-level data valuation," in *International Conference on Networked Computing and Advanced Information Management (NCM)*, 2008, pp. 36-41.

[212] F. F. Moghaddam, M. Yezdanpanah, T. Khodadadi, M. Ahmadi, and M. Eslami, "VDCI: Variable data classification index to ensure data protection in cloud computing environments," in *IEEE Conference on Systems, Process and Control (ICSPC)*, 2014, pp. 53-57.

[213] T. D. Thanh, S. Mohan, E. Choi, S. Kim, and P. Kim, "A taxonomy and survey on distributed file systems," in *Networked Computing and Advanced Information Management, 2008. NCM'08. Fourth International Conference on*, 2008, pp. 144-149.

[214] Y. Cao, C. Chen, F. Guo, D. Jiang, Y. Lin, B. C. Ooi, *et al.*, "ES 2: A cloud data storage system for supporting both OLTP and OLAP," in *IEEE International Conference on Data Engineering (ICDE)*, 2011, pp. 291-302.

[215] A. Adamov, "Distributed file system as a basis of data-intensive computing," in *International Conference on Application of Information and Communication Technologies (AICT)*, 2012, pp. 1-3.

[216] A. P. Deshmukh and K. S. Pamu, "Introduction to hadoop distributed file system," *Journal of IJEIR*, vol. 1, pp. 230-236, 2012.

[217] A. Gómez Cárdenas, X. Masip-Bruin, E. M. Tordera, and S. Kahvazadeh, "A hash-based naming strategy for the fog-to-cloud computing paradigm," presented at the 23RD INTERNATIONAL EUROPEAN CONFERENCE ON PARALLEL AND DISTRIBUTED COMPUTING, 2017, in press.

[218] S. P. Ahuja and B. Moore, "State of big data analysis in the cloud," *Journal of Network and Communication Technologies*, vol. 2, p. 62, 2013.

[219] A. Sharma and P. Gulia, "Analysis of Big Data," *Journal of Analysis*, vol. 1, p. 5372, 2014.

[220] V. K. Jain and S. Kumar, "Big Data Analytic Using Cloud Computing," in *International Conference on Advances in Computing and Communication Engineering (ICACCE)*, 2015, pp. 667-672.