A Data Warehouse for Wi-Fi Urban Data Analysis and Identification of Mobility Patterns

José Francisco de Azevedo Barata
Masters's degree in Networks and Information Systems Engineering
Computer Science Department
Faculty of Sciences of the University of Porto
2024

A Data Warehouse for Wi-Fi Urban Data Analysis and Identification of Mobility Patterns

José Francisco de Azevedo Barata

Dissertation carried out as part of the Master's degree in Networks and Information Systems Engineering

Computer Science Department

2024

Supervisor
Fernando Manuel Augusto da Silva, Full Professor, Faculty of Sciences, University of Porto

External Host Supervisor
Henish Balu, Data Analyst, Associação Porto Digital

Abstract

With the increasing urbanization of the world's population, the topic of smart cities has become extremely widespread. As the population trends to digitalization, being able to harness the potencial of data generated by smart cities is key for the development and quality of life in cities. Wi-Fi, as one of the most widespread technologies, provides a rich source of information for urban management due to the already existing infrastructure in many cities around the world.

Porto Digital Association is a non-profit organization that aims to enable the digital transformation in the city of Porto, providing a citizens a free access network in the city of Porto, consisting of more than 350 APs. This dissertation has the goal of designing and implementing a BI system for Wi-Fi data collected by Porto Digital in order to efficiently leverage the potential of this data.

To this end, this project describes the implementation of the system, from the assessment and improvement of the data collected, to the design and implementation of a Data Warehouse and the ETL process necessary to migrate and integrate data related to Wi-Fi. The final component of this system is an analysis and methods of visualizations for urban applications that aim to provide an understanding of the city for the stakeholders through the lens of Wi-Fi data originating from the integrated data sources.

Resumo

Com o crescimento de urbanização da população mundial, o tópico de cidades inteligentes tornou-se extremamente difundido. À medida que a população tende para digitalização, a capacidade de aproveitar o potêncial de dados gerados por cidades inteligentes é chave para o desenvolvimento de qualidade de vida em cidades. Wi-FI, como uma das tecnologias mais difundidas, fornece uma fonte rica de informação para gestão urbana devido à já existente infraestrutura em várias cidades no mundo.

A Associação Porto Digital é uma organização sem fins lucrativos que procura habilitar a transformação digital na cidade do Porto, fornecendo aos cidadãos uma rede de acesso livre na cidade do Porto que consiste em mais de 350 APs. Esta dissertação tem como objetivo o desenho e implementação de um sistema BI para dados de Wi-Fi colhidos pela Porto Digital de forma a aproveitar de forma eficiente o potêncial destes dados.

Com este objetivo, este projeto descreve a implementação do sistema desde a avaliação de melhoria dos dados recolhidos, ao desenho e implementação de um Data Warehouse e processo de ETL necessário para migrar e integrar dados relacionados com Wi-Fi. A componente final deste sistema é uma análise e métodos de visualização para aplicações urbanas que procuram fornecer um compreendimento da cidade para as partes interessadas através dos dados originários das fontes de dados integradas.

Agradecimentos

Em primeiro lugar quero agradecer profundamente à minha familia e namorada pelo apoio incondicional e motivação que sempre me deram, especialmente nas fases mais complicadas.

Agradeço também ao Professor Fernando Silva pela oportunidade, mentoria, disponibilidade e apoio ao longo de todo o projeto.

A toda a equipa da Porto Digital pela oportunidade, boa disposição e ambiente confortável que me proporcionaram durante o decorrer do projeto, com um especial obrigado ao Balu pela orientação, mentoria e por me indicar pontos de melhoria e encoragamento durante o projeto que foram essênciais para a conclusão do mesmo.

Por ultimo, um obrigado aos meus amigos pela paciência e apoio mesmo nas minhas alturas mais ausentes.

Contents

List of Tables

List of Figures

Listings

Acronyms

AP Wireless Acess Point

NAS Network Acess Server

RADIUS Remote Authentication Dial-In User Service

BI Business Intelligence

DW Data Warehouse

ETL Extract Transform Load

AAA Authentication, Authorization, Accounting

SCD Slowly Changing Dimension

RSSI Received Signal Strength Indicator

CSI Channel State Indicator

SSID Service Set Identifier

CSV Comma-separated values

NIC Network Interface Controller

OUI Organizationally Unique Identifier

OD Origin-Destination

OLAP Online Analytical Processing

FK Foreign Key

MAC Media Access Control

Chapter 1

Introduction

In this chapter, the context and motivations that drive the initiative of the Municipality of Porto, through Porto Digital Association, for this project are presented. Then, the main goals and achievements are discussed and finally an overview of the structure of this dissertation is provided.

1.1 Motivation

Understanding urban dynamics is crucial for developing smarter, more efficient cities. Data collected from Wi-Fi networks offers numerous valuable applications in an urban context and has been an active area of research for years. As the concept of smart cities gains momentum, the importance of Wi-Fi data continues to grow. Studies have shown promising results in various applications, including city user's mobility analysis [28], crowd density estimation [14], characterizing user activity [23] and other diverse applications. These advancements highlight the potential of Wi-Fi data to transform urban environments and improve the quality of life for city inhabitants.

Porto Digital is a private non-profit organization created in 2004, owned by Porto City council, University of Porto and Metro of Porto. Its purpose is to serve as a toolbox for innovation and digital transition for the city of Porto. Among other products and services, Porto Digital provides and maintains a public free Wi-Fi network with more than 350 access points (APs) geographically distributed in the city of Porto. It collects network users' session details for accounting purposes.

Given the network's history and substantial projected growth between 2024 and 2025, as well as the current use of equipment from two different providers, ensuring the integrity of the data collected, and analyzing it to extract relevant information becomes a challenging task. Nevertheless, it presents a valuable source of information for the city of Porto.

This dissertation aims to integrate the available historical and up-to-date Wi-Fi data to facilitate its storage, analysis, and visualization. The goal is to extract meaningful information

for Porto Digital and the Municipality of Porto, ultimately contributing to the city's innovation and digital transition efforts.

1.2 Objectives

The primary objectives of this dissertation are to evaluate and enhance the data collection process, develop a Business Intelligence system that enables the visualization and analysis of the collected data, and apply urban analysis techniques using the resulting system. This involves assessing the current methods of data collection from the Wi-Fi network, identifying and implementing improvements to ensure data integrity, and creating a system for the visualization and analysis of the collected data that is user-friendly and accessible to stakeholders.

Additionally, the project should use the developed system to perform urban analyses, identify patterns and trends within the data, and produce visualizations that aid the Municipality and its departments in making more informed decisions. These objectives aim to leverage Wi-Fi data to provide valuable insights, facilitating urban management and improving the quality of life in Porto.

1.3 Contributions

The main contributions of this dissertation are the following:

- Improvement and standardization of Porto Digital's Wi-Fi data collection system.
- Design and implementation of a Data Warehouse that integrates various Wi-Fi related data sources, facilitating the analysis and accessibility of the for improved decision-making.
- Analysis and creation of visualizations of the integrated Wi-Fi data that help paint a picture of the city through the lens of the users of the network.

1.4 Organization

This dissertation is divided into 6 chapters, structured in the following way:

Chapter 1 - Introduction: A contextualization and motivation of the project is made, presenting the objectives and the contributions.

Chapter 2 - Concepts and Related Work: Technologies and concepts required to understand the rest of the dissertation are introduced in this chapter, more specifically Business Intelligence and Wi-Fi concepts. Lastly, related work on Wi-Fi analysis for urban applications is presented.

Chapter 3 - Data quality, Requirements and Architecture: In this chapter, a description of the network is provided, followed by a data quality assessment. The requirements of the system are then introduced and finally, the architecture of the system is presented.

Chapter 4 - Architechture implementation: This chapter describes the implementation of the design described in the previous chapter, starting with the dimensional modeling and going through each stage of a data warehouse design, ending on the ETL process.

Chapter 5 - Results and analysis: This chapter presents the analysis and results obtained from the data analysis of Wi-Fi data.

Chapter 6 - Conclusions and Future work: In this last chapter the conclusions of the developed project are presented, along with the main challenges encountered during its development. To conclude, future work is discussed.

Chapter 2

Concepts and Related Work

This chapter provides theoretical background for this work, starting by providing BI concepts and its components. It is then followed by Wi-Fi concepts and lastly it presents FreeRADIUS and Authentication, Authorization and Accounting (AAA) concepts.

2.1 Business intelligence

2.1.1 Definition

Business Intelligence, commonly referred to as BI, is an umbrella term popularized by Howard Dressner in 1989 to "describe concepts and methods to improve business decision making by using fact-based support" (Negash et al. [21]).

The field of Business intelligence has grown over the years, and there are several definitions for BI (Chee et al. [11]), and the perspective of BI also changes depending on the applied domain, although commonly it is a data driven decision support system that combines data gathering, Data Storage and data analysis, to provide input to the decision process (Aruldoss et al. [8]).

With the rapid growth of digitalization, organizations have increasingly large volumes of data available, therefore it becomes critical to process and extract meaningful information to aim the decision making process, hence the necessity and importance of BI systems. A study performed by Gauzelin et al. [13] has shown the positive impact BI systems have not only in the decision making process, but also in organization and efficiency of the employees.

2.1.2 Components

Generally, a BI system typically has the following components [8] [24]:

- **Data Sources:** Heterogeneous data sources intended to integrate.

- **Extract Transform and Load:** Transformation and linkage of the different data sources.
- **Data Storage:** Stores historical and current integrated and transformed data.
- **Data Analysis and visualization:** Analysis of the data stored in the Data Storage, using techniques such as OLAP and data mining, usually with a presentation layer to provide users with information in a comfortable and accessible form [24].

Each of these components is further described in the following sections.

2.1.3 Data sources

Data sources, as the name suggests, contain all the data necessary for the Extract, Transform and Load process to be stored in a Data Storage. They can originate from operational databases, spreadsheets, XML files, surveys and many others, and they can be internal to the organization or external. The usefulness of the analysis largely depends on the quality of the data sources and the enrichment they provide.

2.1.4 Extract, Transform, Load

The Extract, Transform and Load (ETL) process consists of 3 steps, and its purpose is to collect and integrate the various sources and finally load them into the Data Storage chosen. Each step is explained in further detail below.

Extract

In this step, various data sources are temporarily loaded to a common area, typically called staging area[Recent Developments in Data Warehousing], so they can be further processed by the transformation step. Since data sources may originate from different systems, having different characteristics, it is essential to effectively manage this step.

Transform

The bulk of the ETL process is usually done in the Transform step. The various data sources may need cleaning, removal of duplicates, data validation, filtering unnecessary data, encryption and anonymization of sensitive data and others. Several integration rules are also applied so that the data integrated data conforms to the desired format in the Data Storage.

Load

In the last step, after the data has been successfully transformed and integrated, it is loaded to the Data Storage chosen now in a format that facilitates the analysis and visualization of the data.

2.1.4.1 Loading strategies

An important step when performing the ETL process, is to identify how the data will be loaded to the Data Storage and the frequency of ETL process, as to keep it up-to-date. Below, the most common loading strategies are described, each with their own application:

Full load

In a full load, all data sources are transformed and loaded in their entirety to the Data Storage. This is typically performed only once, during the first loading of the Data Storage.

Incremental load

With incremental load, also referred to as delta load, data is loaded incrementally at regular intervals, capturing only data with date after the last extraction. This can be new and modified data or data that was not captured with the previous incremental load. There are two types on incremental load:

- **Streaming:** With streaming, data is continuously transmitted from the data sources to the data source, through techniques such as Change Data capture, which propagates any change in a data source to the Data Storage. Streaming is typically used when real-time or near-real-time data is needed on the Data Storage.
- **Batch processing** In batch processing, data is loaded in the Data Storage in predefined chunks, such as daily or hourly data. This is typically referred to as batches.

The selection of a loading strategy depends on the requirements of the BI system, as well as other factors, such as the intensity of the transformation steps and the amount of data processed.

2.1.4.2 Tools

As ETL is an intensive technical process, there are several tools developed for handling the process efficiently. Below some of the most used tools, according to comparative studies from Mukherjee et al. [20], Mali et al. [19], Sreemathy et al. [27] for the ETL process and a review on ETL made by Vassiliadis et al. [29] are presented, along with their advantages and disadvantages.

Informatica PowerCenter

Informatica PowerCenter is a commercial tool designed for the ETL process.

- **Advantages** - Integration of multiple sources and wide range of database connectors, compatible with a majority of platforms [27].
- **Disadvantages** - The tool is complex to use and less efficient at large scale volumes of data due to its resource consumption.

Oracle Data Integrator

Oracle Data Integrator is a product provided by Oracle for fast data integration. It is based on ELt rather than ETL, eliminating the need for a middle staging area between target and source [20].

- **Advantages** - High performance due to its parallel processing and very suitable for high volume tasks . Strong connection to other Oracle Data Warehousing applications [19].
- **Disadvantages** - Focus on integration with oracle databases.

SQL Server Integration Services

SQL Server Integration Services is a component of the Microsoft SQL server for data integration and migration suitable for large enterprises [19].

- **Advantages** - Fast, flexible and easy to use. Very suitable for integrating with other Microsoft products.
- **Disadvantages** - Requires a Microsoft server operating system, limited interaction with non Microsoft sources.

Programming languages

Programming languages such as Python and C are not specifically designed for ETL, but are often used both due to the cost of ETL tools and due to the fact that developers feel comfortable to implement the scripts that manipulate their data by themselves [29].

- **Advantages** - Customization, flexibility, cost and support for the language are the main advantages.
- **Disadvantages** - Programming languages are less user friendly as they do not provide a graphical interface as the tools previously mentioned. They are more complex to use and a very deep knowledge on the language is needed to obtain the same efficiency for large scale data integration and loading.

Airflow

Airflow is a tool for workflow management provided by Apache commonly used to orchestrate ETL and ELT pipelines as Python code. Although not mentioned in the studies presented, it was included for being a good alternative to manage the ETL process when using scripting languages.

- **Advantages** - Free and open source, easy to use and tool agnostic.
- **Disadvantages** - It is not an ETL tool and is not capable of performing integration on its own. The ETL process has to be implemented by a scripting language.

2.1.5 Data Storage

A central component of a BI system is the Data Storage. The three most common types of data storages are the following:

Data Warehouse

The most common Data Storage is the Data Warehouse(DW) [8] [21]. The purpose of the DW is to store large amounts of historical data and current data, resulting from the integrated and transformed data sources, serving as the main source for data exploration and visualization, data mining, business analytics and others [22] to produce visualizations, reports and dashboards that support decision making.

Data Mart

The Data Mart is a smaller version of a DW, in scope and size. It is usually limited to a number of subject areas within an organization. It can be directly built with data from source systems similarly to a DW, or built with data extracted from the DW [32].

Operational Data Store

The Operational Data Store is most commonly used for real-time operational reporting [17]. Real-time data is still integrated, same as the previously mentioned types, but it keeps a very limited amount of historical data with the goal of meeting real-time reporting necessities that can generally not be met with the large amounts of data in a DW or data mart, impacting their performances.

Although Data Warehouses in the context of a BI system are just one of the components, Data Warehousing is also an extensively researched field, hence there are currently several modeling methodologies for the design of a Data Storage, as well as different interpretations of what a Data Storage should provide. In the following subsections, two of the most used DW design approaches, according to Yessad et al. [33] are presented: the Inmon approach [16], and the Kimball approach [17] - along with their advantages and disadvantages.

2.1.5.1 Inmon approach

Inmon [16], defines a Data Warehouse by the following characteristics:

- **Subject-Oriented** The DW should revolve around a particular subject of the organization.
- **Integrated** The sources used for the DW should be properly integrated, meaning it should have a single corporate image such that inconsistencies in the data are no longer present. Inmon enphasizes this point as the most important.
- **Nonvolatile** Data in the DW must not be updated. If there is new data to add to the DW, a new record should be created.
- **Time-variant** Data in the DW must have a temporal component in order to provide an historical view of the data.

Inmon's top-down approach, depicted in Fig. 2.1 in DW design starts from integrating all data sources in a central Data Warehouse that supports all operations for the organization and acts as a single source of truth. Business process specific Data Marts are then built from the Data Warehouse. To build the central DW, data is stored in a normalized state based on the 3rd normal form to avoid redundancy, reducing irregularities at the cost of efficiency.

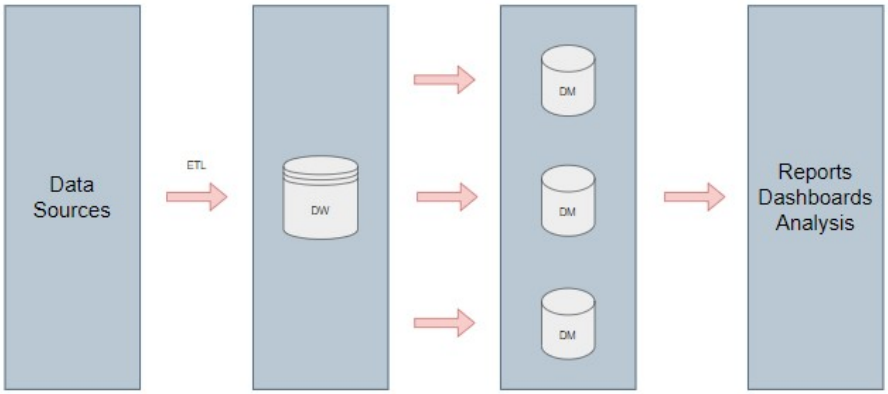


Figure 2.1: Inmon's top-down approach in DW design

- **Advantages** - The data warehouse acts as a single source of truth, reducing inconsistencies in the data. Can handle diverse reporting needs.
- **Disadvantages** - Complex and requires a large initial time investment. Since data is initially stored in a normalized format, queries are slow. The Data Warehouse is only available when all data sources are integrated.

2.1.5.2 Kimball approach

According to Kimball [17], a Data Warehouse should have the following properties:

- It should make information easily accessible.
- It must present the data consistently, meaning it must be credible and the results should be consistent.
- It must be adaptive and resilient to change.
- Be secure to protect organization assets.
- Must serve as the foundation for improved decision making.
- It should be used by the members of the organization, to be deemed successful.

Kimball's approach, also referred to as the Bottom-Up approach, follows the aforementioned principles in its design where Data Sources are integrated in individual Data Marts, which in turn are integrated in a central Data Warehouse, allowing for an incremental implementation of the DW, as illustrated in Fig. 2.2.

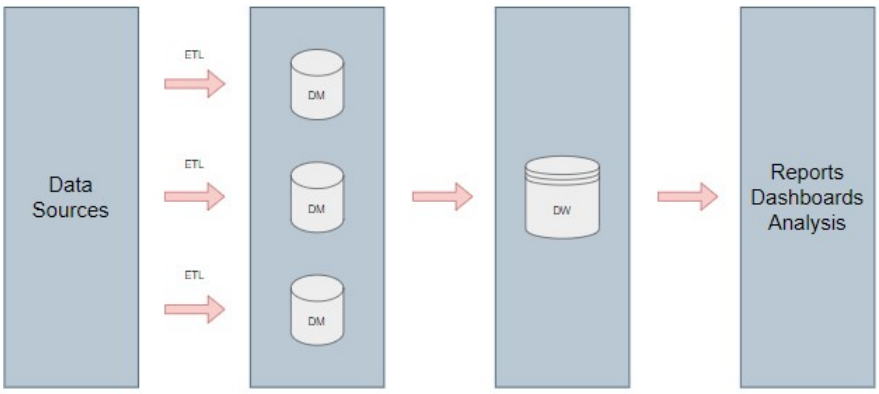


Figure 2.2: Kimball's bottom-up approach in DW design

To model data in a Data Mart, Kimball uses the dimensional modeling approach, keeping the data in a highly denormalized state that offers fast query results and interpretability at the cost of storage.

- **Advantages** - End users can directly use the Data Warehouse as it is built in a easy to interpret format. Offers fast query efficiency due to its redundancy in data. Incremental design means less initial time and resource investment.
- **Disadvantages** - Data is not entirely integrated before reporting. Limited in reporting to the existing Data Marts. Integrating more Data Marts becomes more challenging.

2.1.6 Dimensional Modeling

The dimensional modeling approach in a DW design, popularized by Kimball, has as its main goals understandability and performance, for fast data retrieval and analysis, based on a star

schema design. The key concepts of the dimensional modeling approach are described in the following subsections.

2.1.6.1 Fact table

The fact table is a central concept in a dimensional modeling approach, storing relevant numeric measurements about the business process, and contains two types of columns : measurements, referred to as facts, and Foreing Keys to other dimensions. Since a single row in a fact table is rarely retrieved, the most useful facts are numeric and additive, as they easily allow for summarizing multiple rows when they are aggregated or summed.

2.1.6.2 Granularity

The granularity of a fact table represents the grain at which the facts are measured. Deciding on the grain of the fact table is an extremely important process as the grain will define which facts are stored, and what the dimensions, described in the next subsection, should store. Kimball emphasizes the importance of selecting the lowest granularity possible, as it allows for a higher level of detail of the business process, and that all facts should stay true to the grain selected.

2.1.6.3 Types of fact tables

There are three types of fact tables, derived from the grain at which the facts are stored:

- **Transaction** - Corresponds to facts taken at at a single instant.
- **Periodic snapshot** Corresponds to a summarizing of facts taken during a predefined span of time.
- **Accumulating snapshot** Summarizes facts occurring at predictable steps between the beginning and end of a process.

2.1.6.4 Dimensions

Dimensions give context to a fact in the fact table and store attributes that give more detail to the fact. Dimension tables typically store much less storage than the fact tables. Another key role of the dimensions is to allow for aggregation of data along the dimensions. There are several types of dimensions, and they are presented and described below:

Slowly Changing Dimensions

Slowly changing dimensions (SCD) contain attributes that can change over time. One example of such a case is the address of a person in a dimension that stores attributes related to customers of a retail store.

Since the attributes may change in these dimensions, there are several techniques to handle the changes, such as simply overwriting the value and not keeping record of the changes (**type 1 SCD**), creating a new row on the dimension and adding additional attributes such as the start and end date effective for the record (**type 2 SCD**), which allows to keep an historical record of changes, and finally creating a new column in the dimension, storing the new current value for the attribute changed, while also storing the old value in the same records (**type 3 SCD**)

Degenerate Dimensions

Degenerate dimensions are dimensions keys that do not have a corresponding dimension table, as all the attributes are stored in other dimensions. This type of dimensions are particularly useful for aggregating data.

Date dimensions

Date dimensions are one of the most important dimensions in a DW as it allows for an analysis of the data with a temporal component. It typically stores time related attributes according to the grain of the fact table, such as year, day, hour, minutes, among others.

Outrigger dimensions

Outrigger dimensions are dimensions that contain a reference (foreign key) to another dimension.

Shrunken rollup dimensions

Shrunken dimensions are dimensions that contain a subset of rows or columns of a base dimension, and are required for aggregate fact tables and to capture data at a higher level of granularity.

2.1.6.5 Keys

Keys identify records in the fact table and dimension tables and are used to establish relationships between them, playing a pivotal role in a DW. Each table should have a corresponding key that identifies each record. There are several types of keys:

- **Natural Key** - Naturally identifies a record based on the characteristics of the data, for example the ID attributed by an operational database
- **Primary Key** - Uniquely identifies each record in a table. It may be a surrogate key or a natural key

- **Surrogate Key** - When the natural keys of a production system are not reliable to use, a typical scenario is to replace them by numerical surrogate keys that do not hold any relation to any other attributes [29]
- **Foreign Key** - A key that related to the primary key of another table

2.1.6.6 Schemas

To logically represent data in a DW using a dimensional modelling approach, a schema is necessary, representing the relationships between the different tables. There are 3 most commonly used schemas [10]

Star Schema

In a star schema, a single fact table is connected to all other dimension tables, resulting in a format similar to a star, hence the name. This schema is highly denormalized, resulting in high data redundancy, allowing for efficient queries but do not provide explicit attribute hierarchies and increases storage requirements. Star schema is the most simple and widely used design. Fig. 2.3 illustrates a star schema design for clarity.

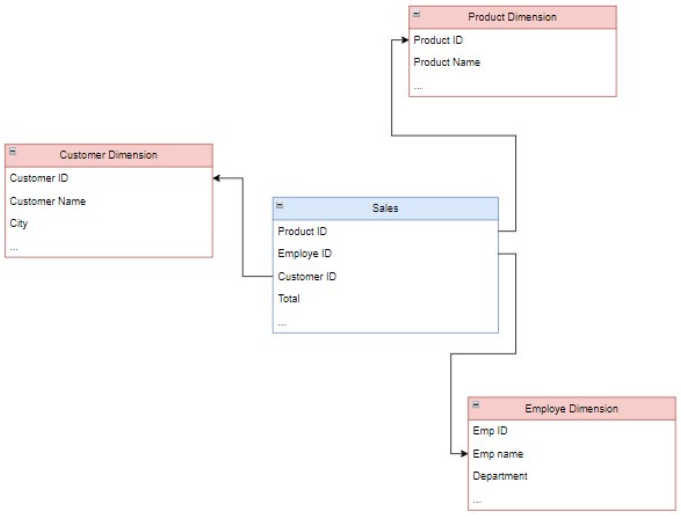


Figure 2.3: Star Schema example

Snowflake Schema

In a snowflake schema, data is still denormalized, but not as much as in a star schema, providing hierarchies for attributes resulting in the normalization of the dimension tables, resulting in less storage requirements but requires additional joins to retrieve data.

Fact Constellation

Fact constellation is similar to the star schema, but allows for multiple fact tables sharing the same dimensional tables.

2.1.6.7 Materialized views

Materialized views [10] are the stored result, in the form of a table, of a query performed on the DW that is often used to store summary and aggregate data or data used in intermediate queries to the DW, providing further processing efficiency and abstraction for accessing it. Utilizing materialized views alows for much of the "heavy lifting" in computation to be performed by the DW such that any application can simply access it, instead of having to perform the heavy lifting itself.

2.1.7 Data Analysis

After the DW is built, there are large amounts of data stored, and the final goal of a BI system is to analyze the data for knowledge discovery and decision making and present it in an easy to understand format. To facilitate complex analysis the most commonly used technique in a BI system is Online Analytical Processing (OLAP), but other techniques such as exploratory data analysis, statistical analysis and time-series analysis for decision making are also utilized [30]

Online Analytical Processing (OLAP) is an important element in a BI system to facilitate complex analysis of multidimensional data, contrary to traditional relational databases that store information in the form of a spreadsheet which in turn makes analysis across different dimensions difficult, OLAP provides a multidimensional view of the data in a DW [10]

At the core of OLAP is the OLAP cube, a multidimensional array-like structure that aggregates data accross multiple dimensions to facilitate analysis and retrieval of information, resulting in an efficient display of high level summaries such as, for example, sales per country or products sold per employee.

OLAP cubes allows for 4 types of operations:

- **Drill-Down** Converts less detailed data into more detailed data by the disaggregation of measures along a concept hierarchy, obtaining a higher level of granularity
- **Roll-up** The opposite of a Drill-Down operation. Aggregates data in a cube along a dimension hierarchy to obtain a coarser granularity
- **Pivot** Rotates the axis of a cube to provide a different view of the data
- **Slice and dice** Splits the original cube, effectively reducing the dimensionality of the data by taking a projection of the data on a subset of dimensions

2.2 WI-Fi

Wi-Fi is a family of wireless network protocols, commonly used for internet access through the use of radio waves by connecting to wireless routers or wireless Access Points (AP). To ensure the proper functioning of Wi-Fi as in to allow for communication between Wi-Fi enabled equipment, signals are sent between devices, which in turn results in several applications resulting from this communication that are very useful to analyze urban dynamics.

2.2.1 Wi-Fi technologies

Wi-Fi probes

User devices that have a WiFi interface periodically perform a wireless probe procedure by actively sending a control frame, known as a probe request. The purpose of this procedure is to have nearby wireless access points send information about wireless networks that are available for connection. This in turn results in APs having access to the Media Access Control (MAC) address of users during this procedure, and the Received Signal Strength Indicator (RSSI) of any user near the AP with an active Wi-Fi to be detected and located.

Channel State Information

Wi-Fi Channel State Information (CSI) records fine-grained information about how a signal propagates between WiFi devices and how a signal is reflected from the surrounding environment in which humans move around. Compared to Wi-Fi probes, CSI can be regarded as Wi-Fi imaging for the environment in which the signal propagates.

Session logs

Session logs refers to logs from users connected to a network for network access. When users connect to a network, the location can be obtained by the AP the user connected to, as well as the RSSI obtained to get an approximate distance from the AP.

2.2.2 FreeRADIUS

As mentioned in 1, PortoDigital collects Wi-Fi session logs of users that use the network. In order to collect this data, a FreeRADIUS server [3] is used. This subsection will provide concepts related to the RADIUS protocol and FreeRADIUS as it will be necessary in order to understand later chapters.

2.2.2.1 Definition

FreeRADIUS is a server implementation of Remote Authentication Dial-In User Service (RADIUS) [26] network protocol based on the Authentication, Authorization and Accounting (AAA) framework. It follows a client-server model and is used in small and large network infrastructures for remote authentication, accounting and access control to different network resources.

The AAA framework is defined as the following:

- **Authentication:** Validates whether the user credentials provided match the information on the AAA server. If the credentials match, the user is granted access to the network, if not, network access is denied
- **Authorization:** Determines what resources the user can access and what permissions the user has on the network
- **Accounting:** Registers information about the user and the session(time the user is connected to the network), after the process of authentication and authorization is completed, until the user disconnects.

2.2.3 Components

To achieve the described above, RADIUS has 3 components that are essential to ensure the AAA process:

- **Network Access Server(NAS):** Also named RADIUS client. It is the gateway between the network and the user. It communicates with the server to authenticate, authorize and pass information about the session. a wireless controller.
- **RADIUS server:** Processes requests coming from the NAS and returns the response so the NAS can apply it. It has no control over what the NAS sends or what it will do based on the response. The server processes the requests coming from the NAS using the contents of the message as well as information it has locally such as SQL or LDAP to decide if a user can have access to the network, or what type of access will it have. FreeRADIUS implements this component of the RADIUS protocol
- **Data Store:** Provides data to a RADIUS server to aid its decision making in authentication and accounting. Typically an LDAP server, directory and/or a database

2.2.4 Session process

The concept of a session is essential in FreeRADIUS and information accounted contains attributes related to a connecting user network usage. Figure 2.4 displays the AAA process and the messages

that happen between the user, the NAS and the FreeRADIUS server. As mentioned previously, the NAS sends an authentication request to the server when a user tries to connect to the network. If the credentials match the user is then authenticated and given access to the network, if not, the connection is denied.

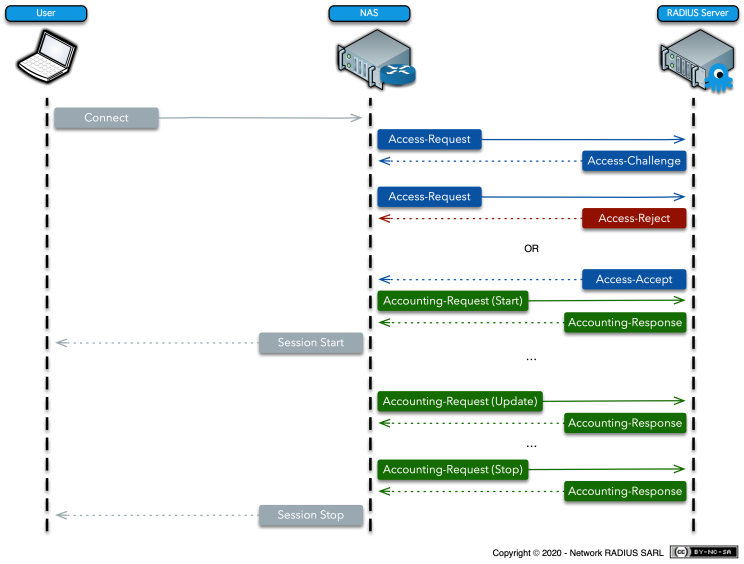


Figure 2.4: FreeRADIUS session process[4]

After the user is granted access, the session formally starts, and accounting begins. In fig. 2.5 accounting in FreeRADIUS is described in detail. The first two rows illustrate the messages between the client and the server, while the last row in the image represents what is added to a directory or database while the messages are happening.

In the first row the message type between the NAS and the server is displayed, which can be a session start, update or stop. Session start represents the beginning of a session and the accounting process, session end represents the end of a session (user disconnection) and therefore the end of the accounting process and finally update is what is sent from the client to the server as the session is kept-alive, to update information about the session. Below it, the content of the messages described above is shown, where some parameters FreeRADIUS keeps accounting of are displayed.

Lastly each message has its respective content added to the database, where the contents of the message for the session are added. The session start results in an insert query, the interim-update in an update query and finally the session stop results in an update query to finalize the session and insert the final session times and stop times for the respective session.

2.2.5 FreeRADIUS session accounting management

In order to correctly identify a session during accounting, FreeRADIUS generates a unique identifier that it attaches to every session: *acctuniqueid*. The generation of this identifier uses a

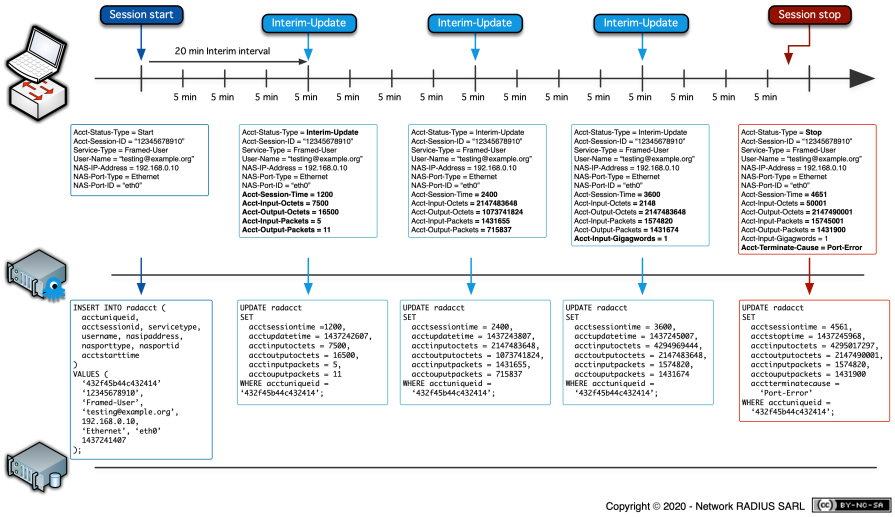


Figure 2.5: FreeRADIUS accounting process[5]

specific formula that may change depending on the version of the tool being used:

- **Version 2** - $acct\_unique\_id = md5(user\_name, acct\_session\_id, nas\_ip\_address, nas\_port)$.
- **Version 3** - $acct\_unique\_id = md5(class\_id, acct\_session\_id)$ if $class\_id$ is supported by the NAS, otherwise it uses the version 2 formula.

This identifier and its generation is fundamental in order to understand how the FreeRADIUS tool interacts with a database and how it stores data about each session for accounting purposes.

2.3 Related Work

After reviewing concepts related to BI, Wi-Fi data and freeRADIUS, this section presents and discusses related works that analyse Wi-Fi data, specifically with the goal of assisting in urban dynamics through the analysis of Wi-Fi data.

A Data Warehouse of Wi-Fi Sessions for Contact Tracing and Outbreak Investigation
To prepare for emergencies such as the COVID-19 pandemic, Zagati et al. [34] seek to leverage existing Wi-Fi infrastructure to assist in large-scale contact tracing.

To this end, the authors designed, implemented and deployed a Data Warehouse for contact tracing using two years Wi-Fi session log data from December 2019 to January 2021 and experimentally deployed it in a university campus in Singapore. This solution integrates the data collected in a staging area, where an ETL process is processes the data and loads it to a Data Warehouse with a multidimensional model extending from the Star Schema. Finally, materialized views are built for more complex and frequently performed queries.

The logical model of the DW developed by the authors, displayed in Fig. 2.6 shares similarities to the dimensional model implemented in this project, apart from the dimensions specifically designed for contact tracing, namely the intervention fact table and the dimensions building, protocol and policy.

Lastly, two visualization and analytics applications are built with support from the DW. The first serves to visualize floor level density throughout the day and can be used to identify crowded regions. The second application depicts all individuals who shared AP location with a target for outbreak investigation and has the option to visualize the trajectory of an infected individual, as depicted in Fig. 2.7. In terms of privacy, user MAC address is hashed during the ETL process.

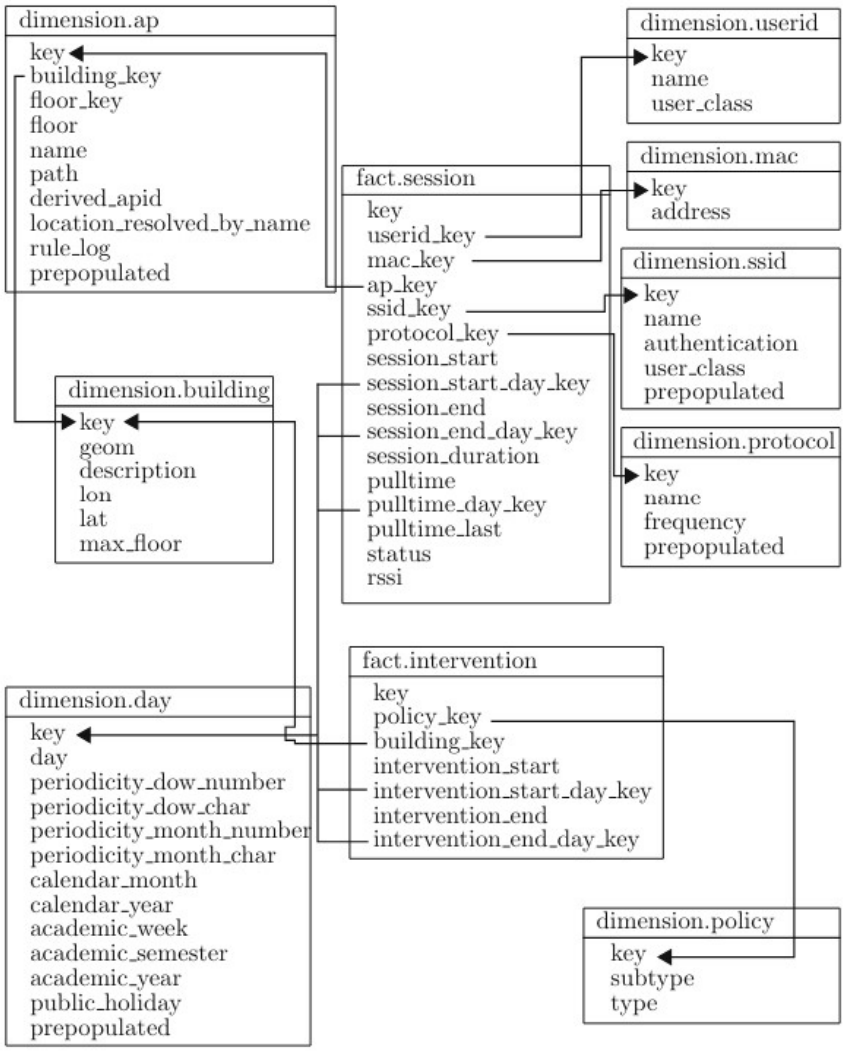


Figure 2.6: Flattened logical model of the DW implemented [34]

Urban phenology: Toward a real-time census of the city using Wi-Fi data
According to Kontokosta et al. [18], effectively modeling population dynamics at high spatial and temporal resolutions would have significant implications for city operations and policy, strategic long-term planning processes, emergency response and management, and public health.

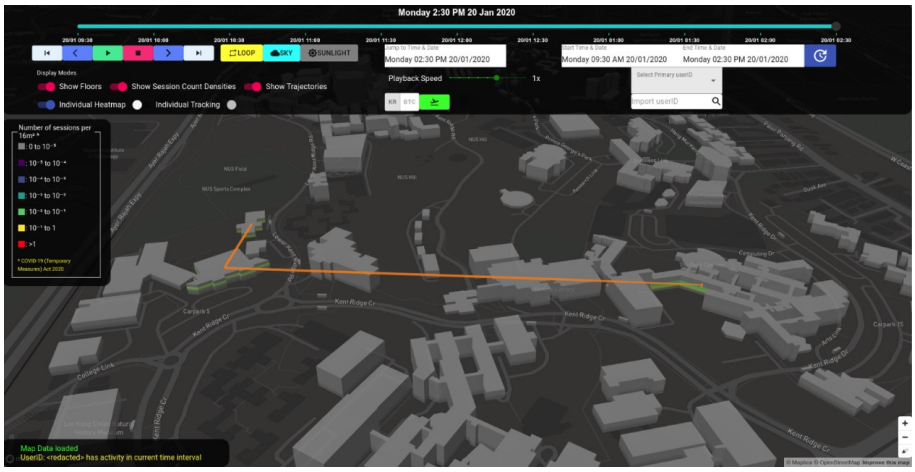


Figure 2.7: act tracing application depicting the trajectory of a target user [34]

With this purpose, the authors develop a model for real-time population estimation by population segment utilizing Wi-Fi Access Point probe data from New York City's Lower Manhattan neighborhood for the period of March to October 2015. Data is obtained through an API from a network management platform, Cisco Meraki, and for privacy concerns, the authors hash the MAC address of the users and only utilize daily and hourly counts.

They first obtain a network baseline and standard deviation of user counts per a pre-classified set of users, namely first timers, daily, occasional and weekly users, displayed in Fig. 2.8 Then, by decomposing the observed hourly patterns for each classification, 3 different populations are obtained: residents, workers and visitors.

Model results of population estimates per population, visualized in Fig. 2.9 are validated against surveys of populations counts for the area in study, obtaining results within 5% of the survey counts for residential population and within 2% for the worker population.

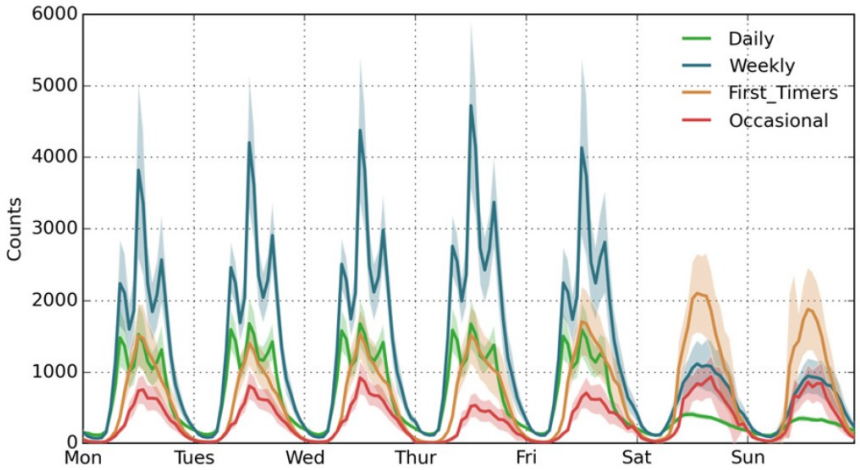


Figure 2.8: Mean counts per hour by user group [18]

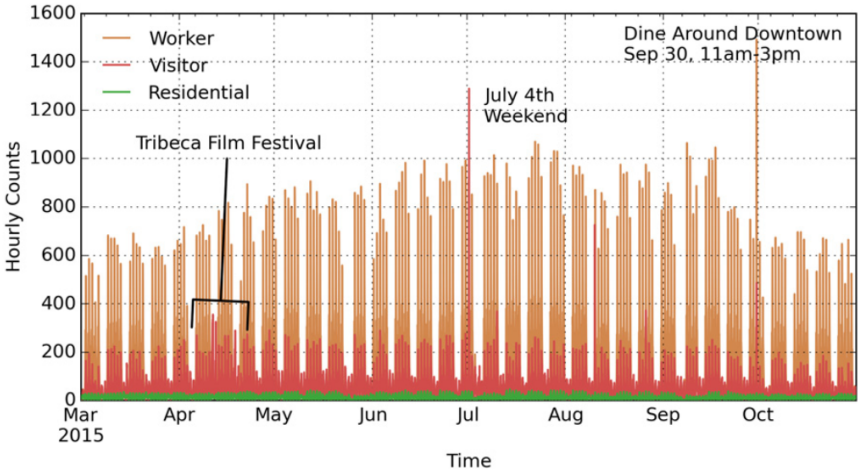


Figure 2.9: Model population estimates obtained [18]

Wi-Fi based city users' behaviour analysis for smart city

Belini et al. [9] states that monitoring, understanding and predicting city user behaviour is one the major topics in the context of Smart City management. As such, it is mandatory to develop methods and tools for assessing people behaviour in the city.

For this purpose, the authors develop a methodology to instrument the city, via the placement of APs, to use them as sensors to capture and analyze urban behaviour. The methodology developed is used to select suitable APs in the city of Florence, Italy, to collect Wi-Fi Access Point probe data for the period of May to December 2016.

After selecting suitable APs, they analyse the flows of people in the city by computing an Origin-Destination matrix (OD) using zone clusters as origins and destinations in the city. To represent and analyse the OD matrix, an application for visualizing the flows is developed, shown in Fig. 2.10. The application developed displays the inflows or outflows for a selected zone cluster for several periods of the day and zooming in and out on the map redraws the zone clusters, aggregating the flows between the various zones.

To understand city usage, the trend of each AP is obtained from the average value per time slot, and the trends obtained are clustered using different clustering techniques to identify APs with similar usage, identifying 12 different clusters. Lastly, a predictive model using auto-regressive integrate moving average approach is built to predict access point connections. The autoregressive part the of model creates the basis of the prediction, which is improved by a moving average modelling for errors made in previous time instants of the prediction, showing positive results in predicting AP connections

Digital footprints: Using WiFi probe and locational data to analyze human mobility trajectories in cities

"City governments all over the world face challenges understanding mobility patterns within

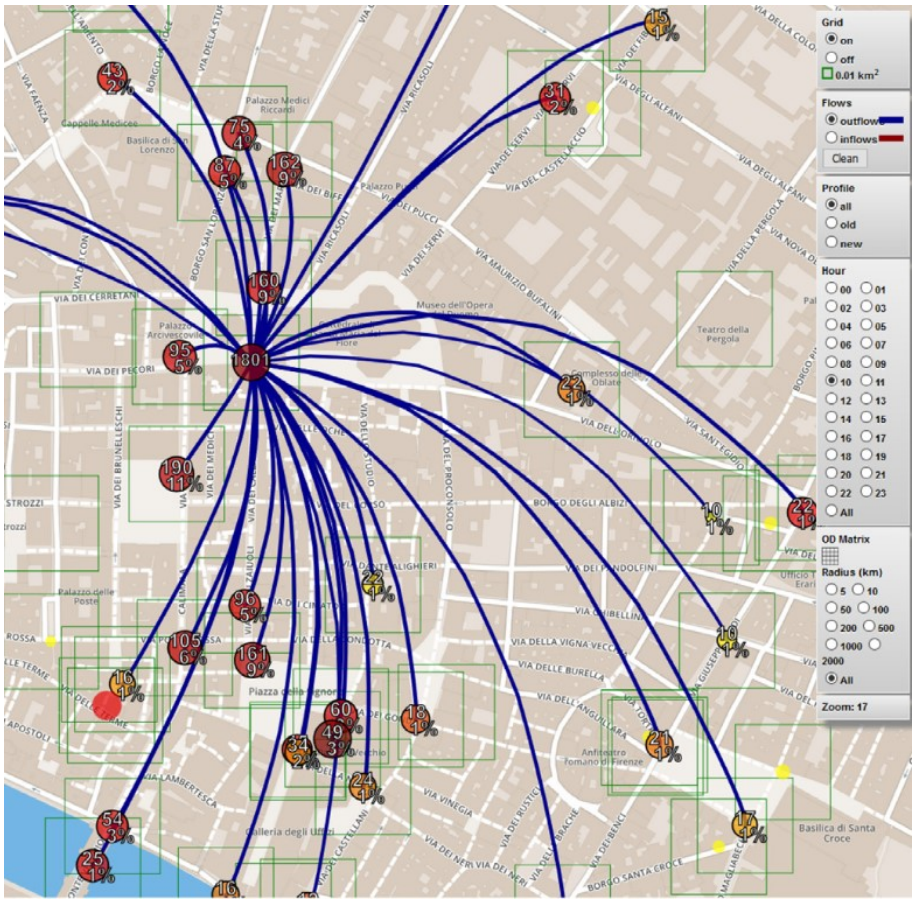


Figure 2.10: Application to analyse the OD matrix [9]

dense urban environments at high spatial and temporal resolution" [28]. As such, the authors model urban mobility trajectories using spatial network data from public Wi-Fi Access Point probe data collected in New York City for the period of one week, in order to demonstrate the potential of Wi-Fi data.

To anonymize data, user MAC addresses were replaced by an incremental identifier starting from 1, and the original MAC address was deleted. The authors first aggregate data per AP and per day and compare it to manual counts in the area to demonstrate the usefulness of Wi-Fi data for population estimates. Then, to identify spatial variance, user counts per AP and day are plotted on a map to identify areas with high populational count.

Lastly, to model mobility, client activities are converted to a graph where nodes represent APs and edges represent flows between consecutive AP entries in a journey. To analyze mobility from the graph, graphs for different days and periods are presented and discussed. The image in Fig. 2.11 visually displays the network graph obtained for the weekday mornings for the study period, where the edges represent the flows of users and the blue arrows represent the destination of the flows.

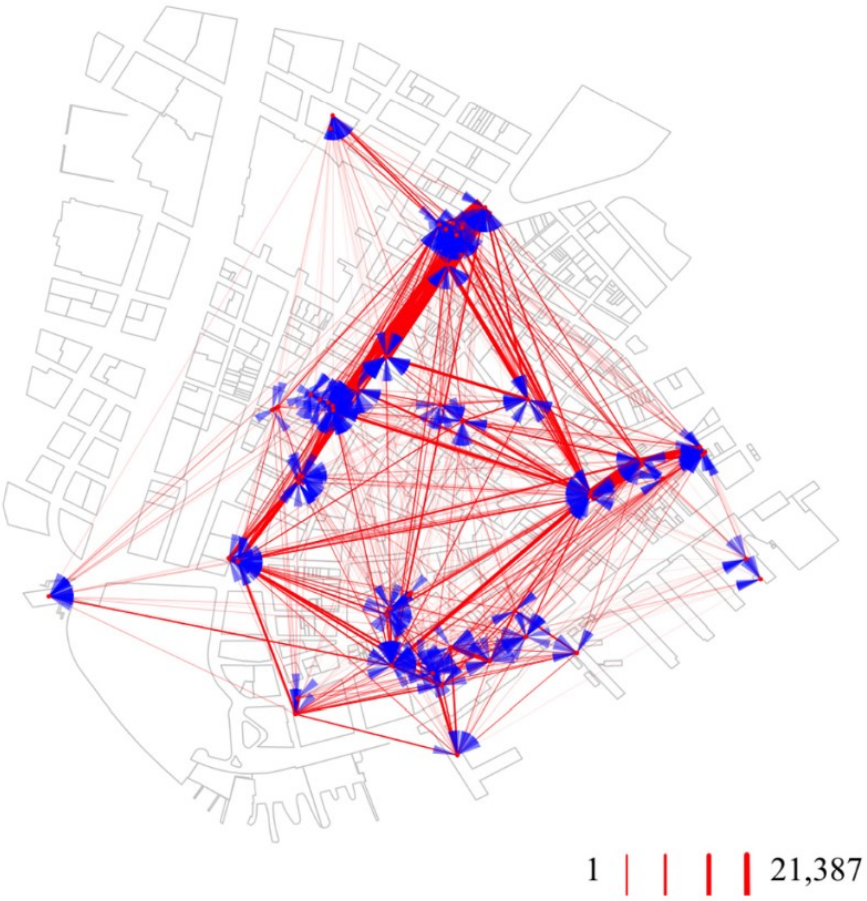


Figure 2.11: Network graph with destinations of movements [28]

Chapter 3

Data Quality, Requirements and architecture

This chapter provides a description of the network and the collection system followed by the requirements of the BI system. Data quality is then assessed and solutions are provided for its improvement. Lastly, the architecture defined for the system is presented.

3.1 Network and Infrastructure description

The network consists of more than 350 APs in several locations in the city. The APs may broadcast 4 different Service Set Identifiers(SSID), of which two are for user access:

- Porto.FreeWifi - A public access network provided to citizens and visitors of the city, requiring no login information and free to use.
- Eduroam - An account-required network, used by students and researchers, also free to use. Through a protocol with the University of Porto, APs may provide Eduroam access.

Users accessing the network go through a captive portal and after agreeing to the terms of service they are granted internet access.

Figure 3.1 illustrates the layers involved in the process from a user initiating a session and the accounting of the session data. A user first connects to an AP in the AP layer, which communicates with one of several NAS (FreeRADIUS clients) in the NAS layer to authenticate the user. The NAS then communicates with a FreeRADIUS server in the server layer, that may grant access or reject the user. If the user is granted access, the information is sent back through the layers to the AP where it starts providing internet access. At the same time, the servers in the server layer begin accounting information to the operational database layer. In case of failure, there are redundancy servers in all layers except the AP layer to guarantee that the process can continue without the loss of data.

The infrastructure supports an immersive connection experience, meaning that when a user moves from one AP to another in the same session, the user does not have to reconnect to the network.

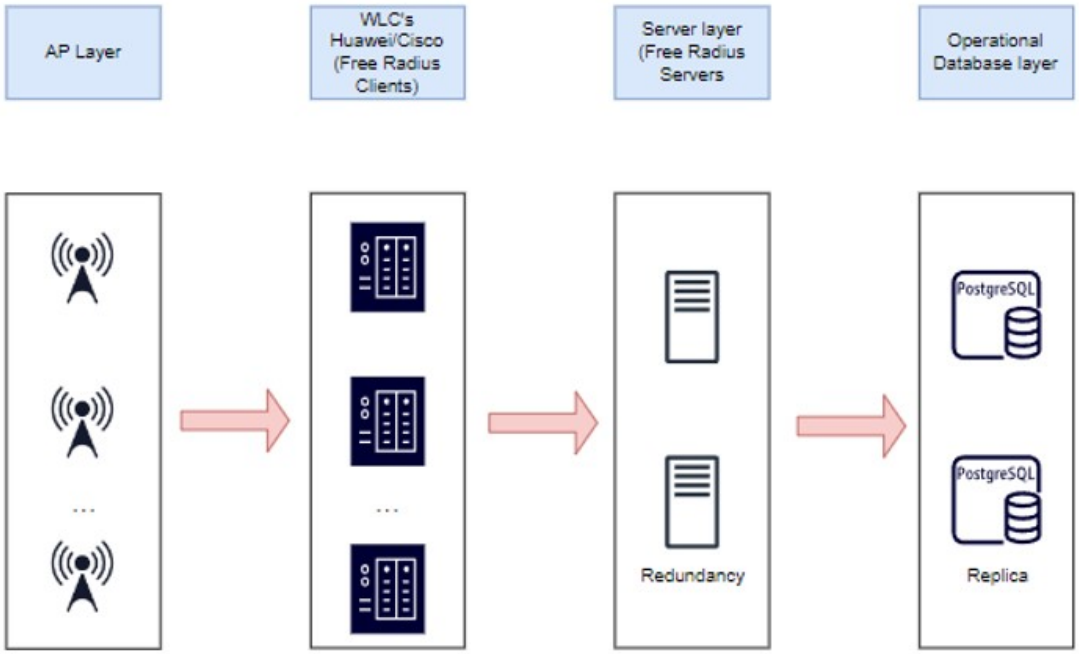


Figure 3.1: Layers of the infrasctucture for Wi-FI AAA in PortoDigital

3.2 Data Quality

3.2.1 Previous solution

A first approach to build a BI system from the collected data was previously attempted by Porto Digital, with the architecture shown in Fig. 3.2, where data was migrated directly to another SQL database and transformed into a dimensional model using the database as a staging area. This attempt had the assumption that session data was fixed in the operational database, which did not work as an issue where sessions records in the operational database that had already ended were being constantly updated (rewritten) to more recent days. This interfered with the ETL process and also highlighted problems with session integrity in the data collection process, hence the necessity to reevaluate it.

3.2.2 Problem description

To evaluate the data collection system and data quality, the first step taken was a validation of the problem described in the previous subsection. Figure 3.3 shows the count of records per day, based on the time the session was first initiated, for the period of 31st December to the 18th of

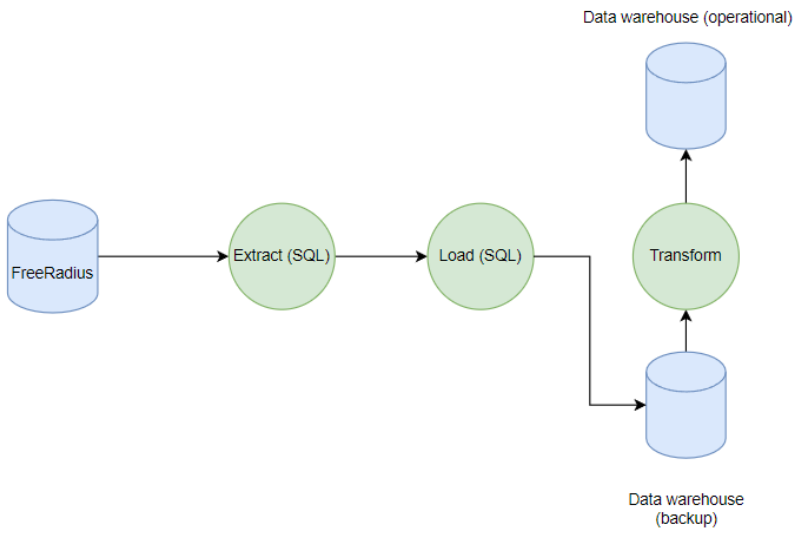


Figure 3.2: Previous architecture for the ETL process

January, taken from the operational database on two different days: 22th of January and 25th of January.

It would be expected that the number of sessions from previous days would remain constant as the session start time was utilized to obtain the count. As such, there should be the same number of sessions in the operational database for the period of 31st of December to 18th of January when the count is taken after this period, but this was not the case as can be observed in the figure by the large difference in the number of records, displayed as the red number on top of the bars. Just for the 18th of January alone, more than 3000 session records where already missing when comparing the counts taken at the 22th to the counts at the 25th of January for this period, indicating that historical records were indeed not constant, which severely impacts the integrity of the collected data and poses challenges towards designing and implementing a BI system for Wi-Fi data.

The missing records for the period were searched in the operational database by its unique identifier and they were confirmed to be updated to more recent days, with the start time of the session and other attributes altered.

3.2.3 Data quality analysis

To address these challenges, a descriptive analysis of the data was performed, as well as an in-depth study of the FreeRADIUS tool accounting process and how it is configured in Porto Digital infrastructure.

When using a session accounting tool based on the Radius protocol, as mentioned in chapter 2, one of the most important attributes for accounting is the session identifier. This is generated by the NAS and is required to be unique for all ongoing sessions [26]. FreeRADIUS does not

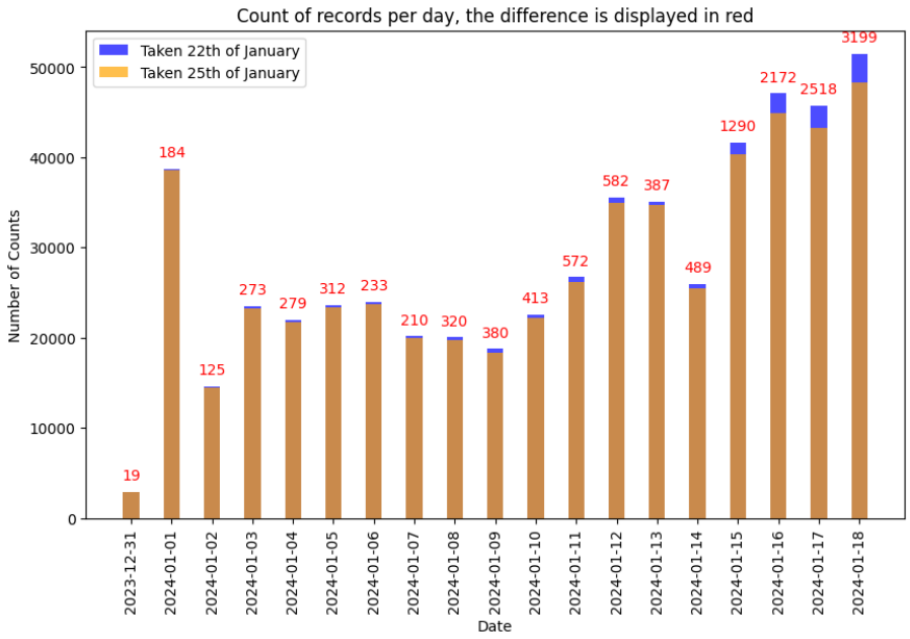


Figure 3.3: Number of rewritten sessions for a past period in the operational database

rely on the session identifier alone as it may repeat, so it uses additional attributes to ensure the uniqueness of a session such as attributes of the NAS and username to account session data. As such, in order for a session to be properly managed in FreeRADIUS, any of the following four attributes should be present to ensure the uniqueness of the session: session identifier, username, NAS port and finally NAS IP address, as previously mentioned in chapter 2 for the v2 version of the tool, which is the currently used by PortoDigital.

In the case of PortoFreeWifi SSID, which is an open free Wi-Fi network, a username is naturally not provided by the user, but it still is important for session management. Thus, for PortoFreeWifi sessions, for each NAS, the user MAC address should be placed in the username field in order to aid in uniquely identifying a session.

In Porto Digital infrastructure, there are 7 active NAS, responsible for specific APs. Further analysis of the updated sessions described in the previous section shows that all altered sessions identified in Figure 3.3 originate from only 3 NAS, all of them with SSID PortoFreeWifi. 60% are from one NAS (A), which is also the NAS with the most sessions in general, NAS B is responsible for 37%, and finally NAS C the rest 3%, as shown in Fig. 3.4

By analyzing missing values for the most relevant attributes for each NAS, the cause of the problem became clear: for PortoFreeWifi sessions, the FreeRADIUS server, was not receiving a session identifier from NAS A and C. For NAS B, the session identifier was in the smallest format possible (8 bytes), resulting in it being prone to repetition, and the username received was empty.

The impact on data quality is large. For both NAS with no session identifier, A and C, each time the same user connects to the same port, the session data will be updated for some

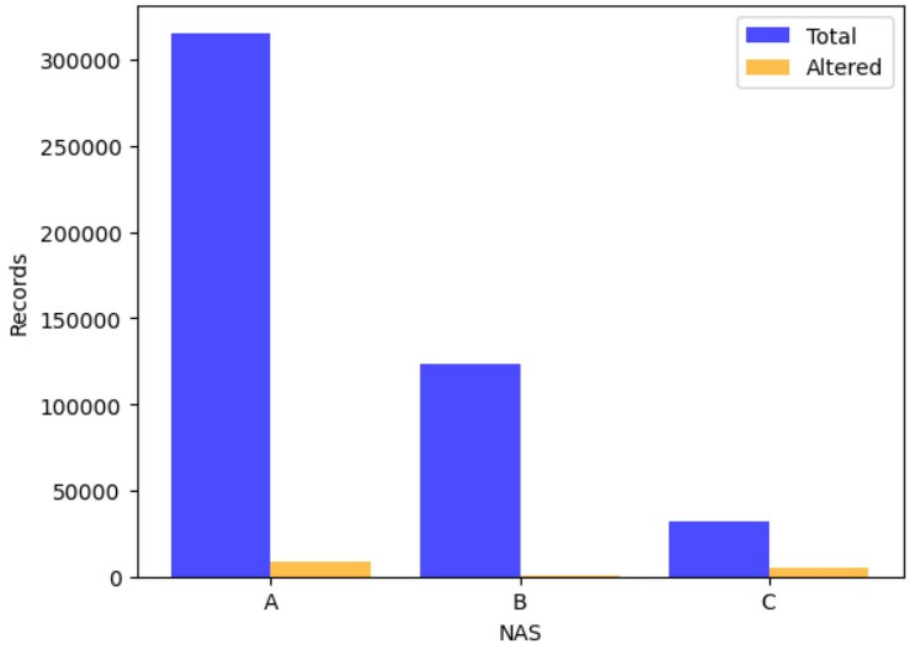


Figure 3.4: Altered records vs total records

attributes. The previous data for the record is lost and the new session information is also not correct as not all attributes are updated until the session is over. This also results in fewer overall sessions for these two NAS, as the sessions will be rewritten instead of creating a new record.

For NAS B with a smaller session identifier and no username, the impact on data quality is not as significant, as it will only update sessions if the session identifier happens to be the same for the same port. Given that the session identifier is in the smallest format possible (8 bytes) it is still likely to happen, which results in the 3% observed even though it has more sessions than NAS C.

The remaining NAS did not have large issues as all the important attributes were present and the session identifier had the largest format possible, 32 bytes, ensuring the integrity of the sessions.

The solution provided is a standardization in the configuration of every NAS to utilize a 32-byte session identifier for all SSID's and the user MAC address as a username for PortoFreeWifi sessions. The session identifier alone is not enough as it can eventually repeat, so the formula used by the FreeRADIUS tool should still be kept to ensure session uniqueness. Lastly, the operational database should only keep at most 6 months of historical data (the period required for these systems by law to store data), to further minimize collisions between identifiers. The mentioned solution was discussed the infrastructure team and data analysis team and implemented.

Another issue was noticed during this analysis - a large number of sessions with no stop time attributed, mostly on NAS A and B, which are both responsible for APs with the most PortoFreeWifi sessions. Looking at the relevant attributes for session management, the NAS port identifier is included. This means that when a user connects to a different physical port on the

NAS, for example, changing AP, the unique session identifier also changes, causing the session to split between multiple records while ongoing. This inflates the total number of sessions, but also provides additional information about user location as if it were not the case, only the last AP to which the user connects to would be kept.

Since these records provide information about the mobility of users, a decision was made to keep all records as they will be very useful for analysing urban dynamics.

Some other small issues were found, which were also mentioned in [34] such as session stop times a few milliseconds earlier than start times and session times not calculated correctly. There were also cases of values of upload and download on the maximum value allowed, which were likely to be outliers due to the short session time.

These quality issues were not feasible to be solved by the infrastructure's team, so they are taken into account later on in the ETL process, described in Chapter 4.

Some APs were being accounted for in the FreeRADIUS server but were not present on the list of active APs, a CSV file with information on all active APs. This information was sent to the infrastructure team for updating.

The standardization in configuration provided and the architecture designed and implemented in the following chapters may be useful for any organization that already use the FreeRADIUS tool and/or intends to perform large scale analysis on Wi-Fi data for both account required and free access networks while also making use of the rich mobility records it provides.

3.3 Requirements

The solution developed for this project must meet the following requirements:

- Take into account data quality problems, ensuring the integrity of the sessions
- Integrate all data sources relevant to the analysis of Wi-Fi data
- Allow for historical analysis of the data
- Have a fast query performance for dashboards and analysis
- Allow for analysis at different grains, such as daily and monthly
- Take privacy concerns into consideration as to not infringe on the privacy of users

3.4 Architecture design

The proposed architecture for the BI system is also structured in layers, as illustrated in Fig. 3.5

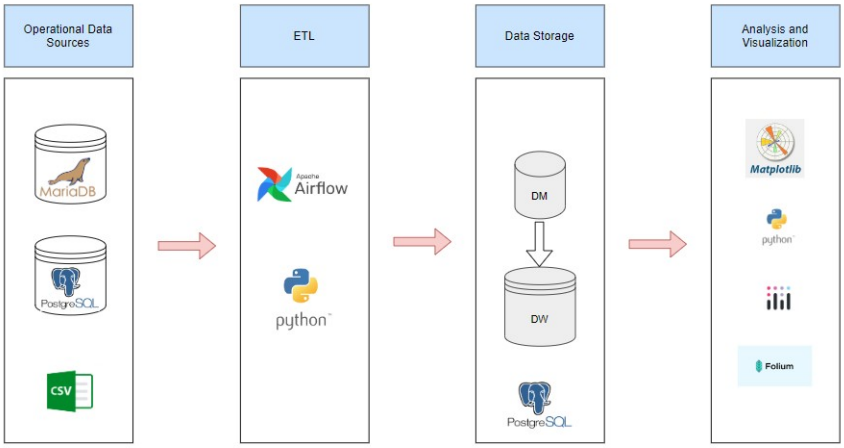


Figure 3.5: Proposed System Architecture for the BI system

Data sources

The first layer of the system are the data sources, since they provide the data necessary for the remaining layers of the architecture. The data sources are utilized are described below:

- A Maria DB database with historical FreeRADIUS Wi-Fi session data.
- A PostgreSQL database with both historical and operational FreeRADIUS Wi-Fi session data
- CSV with all active PortoDigital APs and their respective information
- CSV with Eduroam domains and their respective information
- CSV with NAS information

Data from the databases correspond to raw data captured by the collection system using FreeRADIUS. The other data sources are complementary data that enrich the information related to the sessions.

ETL process

The second layer of the architecture is the ETL process. In this step, data is extracted from the data sources, performing the necessary transformations with the Pandas library to the format of the dimensional model designed, and finally loading them to the Data Warehouse.

The Airflow tool is used for orchestrating the pipeline that implements the ETL process and Python is used for its implementation. Python was selected for the implementation due to previous knowledge of the language, its flexibility in handling the transformation process and finally no cost of use. Airflow was selected for orchestration of the pipeline for also being free to use, and for being a very suitable tool for orchestration when using python scripts.

Data Warehouse

After the ETL layer, the next layer is the Data Warehouse. Data is stored in a Postgresql database, more specifically, since a bottom-up approach was selected, data is stored in a Data Mart that constitutes the first Data Mart for Wi-Fi data in PortoDigital. Hence, it will be simply referred to as the Data Warehouse.

Analysis and visualization

The last layer is the analysis and visualization of the data after it is is fully loaded and integrated in the data warehouse. Python libraries were used for this layer. The technologies and methods utilized will be further discussed in Chapter 5

Chapter 4

Architecture Implementation

This chapter describes the implementation of the BI system, detailing each implementation step, starting with the dimensional model chosen, followed by the pre-processing of the data sources. Finally, the ETL process is described.

4.1 Dimensional modeling

After reviewing BI concepts and design approaches, related work as well as the requirements of the Data Warehouse, Kimball's bottom-up approach was selected due to time limitations and shorter implementation times, query efficiency offered by dimensional modeling and ease of use.

The dimensional model, displayed in Fig. 4.1, follows a star schema design with a single fact table, having the lowest granularity possible to maintain a high level of Wi-Fi session detail and flexibility, as strongly recommended by Kimball [17]

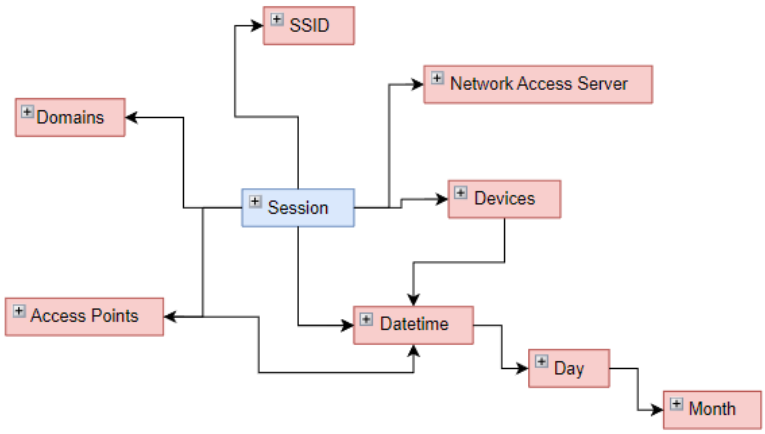


Figure 4.1: Dimensional model for the Wi-Fi sessions Data Warehouse

4.2 Fact table

As mentioned in the previous section, there is a single sessions fact table that connects to all other dimension tables, except for the shrunken dimensions. It contains attributes about each FreeRADIUS session, with the finest granularity possible - session level granularity. Table 4.1 describes each attribute in the session fact table.

Attribute	Description
`sessionid`	Unique ID that identifies each record in the DW
`acctuniqueid`	FreeRADIUS unique session ID
`acctsessionid`	session identifier from the NAS
`device_id`	ID from the devices dimension
`ap_id`	ID from the access_points dimension
`nas_id`	ID from the network_access_servers dimension
`domain_id`	ID from the domains dimension
`ssid_id`	ID from the SSID's dimension
`start_date_id`	ID from the Dates dimension
`stop_date_id`	ID from the Dates dimension
`download`	Amount of download during the session in megabytes
`upload`	Amount of upload during the session in megabytes
`sessiontime`	Total session time in minutes

Table 4.1: Sessions fact table attributes and description

Session records are uniquely identified by the `sessionid`, a surrogate key that increments for each session record. An incremental surrogate key was chosen for several reasons:

- There are two different sources for FreeRADIUS data, each with their own natural keys, `acctuniqueid`, that may result in collisions when merging the data from both data sources.
- As shown previously in Chapter 3, collisions may still occasionally occur even for the same data source, hence, an incremental primary key ensures that even if a session record is updated in the operational database, both records are kept in the DW.

4.3 Dimension tables

The model consists of 8 dimension tables, two of which are shrunken dimensions, and also makes use of materialized views, to create cubes for OLAP and to allow for more efficient analysis of relevant data.

Access Points

The Access Points dimensions is a type 2 slowly changing dimension and contains all attributes of the APs linked to each session. The attributes and respective description are shown in Table 4.2.

Since APs may change location there is a need to keep an historical record of the locations the AP was in so during analysis it correctly correspond to its location at the time. As such, the attributes `created_at`, `end_date` and `status` where added to keep this historical information.

When an AP changes location, a new record is created and sessions point to the most recent record available for each AP, verified by the `first_time_logged` attribute which is a foreign key connecting to the Datetime dimension.

Attribute	Description
`ap_id`	Unique ID identifying each AP
`called_station_id`	AP MAC address
`hotspot`	Hotspot the AP belongs to
`name`	Technical name of the AP attributed by the infrastructure team
`latitude`	Latitude of the AP location
`longitude`	Longitude of the AP location
`zone_number`	Zone number where the AP is located
`zone_name`	Zone number where the AP is located
`parish`	Parish where the AP is located
`street`	Street where the AP is located
`in_out_door`	Flag indicating whether the AP is located indoors or outdoors
`vendor`	Manufacturer of the AP
`status`	Indicates if the AP is active or non-active
`municipal_company`	Municipal company the AP is providing service
`description`	Concrete location of the AP
`created_at`	Indicates the date the record was created
`end_date`	Indicates the date the record was out of use

Table 4.2: Access Points dimension table attributes and description

Network Access Server

The Network Access Server dimension contains all relevant attributes of the NAS and its attributes and respective description are displayed in Table 4.3.

Since the IP address is fixed for each NAS, `nas_id` is a natural key based on the IP address.

Attribute	Description
nas_id	Unique ID that identifies each NAS
nas_ip_address	IP address of the NAS
vendor	Manufacturer of the NAS
pop	Point of presence of the NAS

Table 4.3: NAS dimension table attributes and description

SSID's

The SSID's dimension table contains information about the SSID of each session and has two attributes only, displayed in Table 4.4. The natural key ssid_id, based on the SSID of each session, and the respective SSID of the session such as Eduroam, PortoFreeWifi and others.

Since there are more than two SSID's it is not easily represented by a flag measure in the fact table and as relevant information may be added later specific to SSID's, a dimension table was created.

Attribute	Description
ssid_id	Unique ID identifying each SSID
ssid	The respective SSID of the session (Eduroam, Porto.FreeWifi, ...)

Table 4.4: SSID's dimension table attributes and description

Devices

The Devices dimension table contains the relevant attributes of the devices that connect to the network. The attributes, described in Table 4.5 are the most sensitive piece of information stored.

As such, both the calling_station_id and username are anonymized. The anonymization procedure of these two attributes will be described in detail in the implementation of the ETL process.

The first_time_logged attribute is a foreign key to the datetime dimension that stores information about the date the device was first seen.

Domains

The Domains dimension table contains information specific to Eduroam sessions as this information is processed from the institution information contained in the institutional email contained in the username.

Attribute	Description
device_id	Unique ID that identifies each device
calling_station_id	Anonymized device MAC address
username	Anonymized username for Eduroam sessions
first_time_logged	Indicates the date the device was first seen

Table 4.5: Devices dimension table attributes and description

The attributes, displayed in Table 4.6 contain all the relevant about the domains, namely the university, country and college that the email part of the username belongs to.

Attribute	Description
domain_id	Unique ID that identifies each domain
domain	Domain of the Eduroam session
short_name	Short name of the College
full_name	Full name of the College
university	University the user belongs to
country	Country of the university

Table 4.6: Domains dimension table attributes and description

Datetime

The Datetime dimension contains information about the time and date each session occurred and is the largest dimension table since it has a granularity down to the minute, as can be seen in Table 4.7.

The date_id is the natural key of the table, derived from the combination of date and time. It includes a foreign key to the days dimension with the attribute day_id, and also serves as an outrigger dimension for both the Access Points and Devices dimensions trough its natural key.

Days

The days dimension is a shrunken dimension of the datetime dimension, made to allow for aggregate materialized views and to capture data at a higher level of granularity. Table 4.8 describes all attributes contained in the dimension, which describe a day in a much finer detail than the datetime dimension, adapted for automated reporting. It also contains a foreign key, month_id that connects to the months dimension.

To provide an example usage, when aggregating the sessions at a daily granularity, to obtain for example the total number of sessions per day in a month, the sessions would have to be

Attribute	Description
date_id	Unique ID that identifies each datetime in the DW
timestamp	Timestamp the session occurred in
year	Year the session occurred in
month	Month the session occurred in
day	Day the session occurred in
hour	Hour the session occurred in
minute	Minute the session occurred in
week	Week the session occurred in
day_type	Type of day the session occurred in (Weekend, Week)
week_day	Name of the day of the week
date	Date part of the datetime
day_id	FK to the days dimension

Table 4.7: Datetime dimension table attributes and description

aggregated by the days and the respective month attributes from the datetime dimension, but there would be no further information about each of them as they would not have their specific dimension.

With the days dimension, and for the same purpose the months dimension, aggregation at a daily granularity can be simply made using the day_id attribute and to extract relevant information about each day, since the data was aggregated by this attribute, it can simply be joined back to its respective dimension.

This also allows for the creation of materialized views with aggregates at a daily granularity that then connect to the days dimension.

Typically, a shrunken dimension contains a subset of columns from the base dimension, but this approach was not chosen, as the datetime dimension has a high level of granularity, as previously mentioned, tending to be quite large. As such, more detailed information about days and months are left to their respective dimensions instead of placing all the information on the datetime dimension, substantially reducing its size.

Months

The Months dimension, as previously mentioned, is a shrunken dimension to allow for aggregate views at a yearly and monthly granularity, hence there is a high level of detail for attributes specific to these two granularities.

Its attributes, visualized in Table 4.9, are designed with the same concept as the days dimension. Month_id is a natural key derived from a combination of year and month to

Attribute	Description
day_id	Unique ID that identifies each day
date	Date part of the Datetime
year	Year the day belongs to
month	Month number
month_name	Full month name
day_of_month	Day of the month
month_id	FK to the Months dimension
day_of_week_num	Number of the day of the week
day_of_week_name	Full name of the day of the week
day_of_week_name_short	Short name of the day of the week (mon, sat,..)
week_of_month	Number of the week of the month
week_of_year	Number of the week of the year
day_type	Indicates whether it is a weekend or a weekday

Table 4.8: Days dimension table attributes and description

uniquely identify each month of each year, and all attributes provide a high detail of months and years.

Attribute	Description
month_id	Unique ID that identifies each year month combination
month_name	Full name of the month
month_name_short	Short name of the month
quarter	Quarter the month belongs to
year	Year the month belongs to
year_quarter	Year quarter combination the month belongs to
year_month_name	Year month combination the month belongs to
first_day_of_month	First day of the month
last_day_of_month	Last day of the month

Table 4.9: Month dimension table attributes and description

4.4 Materialized views

To support dashboards that utilize the developed DW for reporting and analysis, and make the retrieval of data at a higher granularity more efficient, three materialized views were created. Two different OLAP cubes were made using materialized views and a normal materialized view

with raw data. The data source for the materialized views is the DW.

Monthly yearly aggregate

The Monthly yearly aggregate is an OLAP cube in the form of a materialized view with data aggregated by all dimensions, the `month_id` and a sum of the fact measures, as shown in Table 4.10, essentially functioning as a fact table with a monthly granularity. This allows for efficient analysis of monthly totals on all dimensions and supports a dashboard for historical data.

This materialized view contains totals per aggregation instead of the measures observed in the fact table, namely the total session time, total download and upload, and session counts.

Attribute	Description
`ap_id`	ID from the AP dimension
`ssid_id`	ID from the SSID dimension
`domain_id`	ID from the Domains dimension
`nas_id`	ID from the Network Access Server dimension
`device_id`	ID from the Devices dimension
`month_id`	ID from the months dimension
`total_sessiontime`	Total session time for the aggregation
`total_download`	Total download for the aggregation
`total_upload`	Total upload for the aggregation
`session_count`	Session count for the aggregation

Table 4.10: Monthly yearly aggregates table attributes and description

Daily aggregate

Similar to the monthly yearly aggregate view, the daily aggregate view is an OLAP cube containing two years of aggregate data for all dimensions, day_id and sums of the fact measures. It also supports a dashboard for historical data. The attributes are the same as the monthly aggregate view, except that day_id is included instead of month_id, functioning as a fact table with daily granularity.

Biblio data

The biblio data view supports a dashboard with data specific to sessions initiated daily in a municipal library. This is a service provided to the municipal library, hence a materialized view was created to support the dashboard in efficiently retrieving this data.

The attributes, visualized in Table 4.11, are similar to the sessions fact table, except that it only includes ap_ids that are in the library and it is connected to the days dimension instead of the datetime dimension, since it requires daily sessions.

Attribute	Description
ap_id	ID from the AP dimension
ssid_id	ID from the SSID dimension
domain_id	ID from the Domain dimension
nas_id	ID from the NAS dimension
device_id	ID from the Device dimension
day_id	Id from the Days dimension
session_time	Session time of the session in minutes
download	Download of the session in Megabytes
upload	Upload of the session in Megabytes

Table 4.11: Biblio data table attributes and description

4.5 Data Sources

The next step was to prepare the data sources for the ETL process. As previously mentioned in Chapter 3, data sources are integrated into the DW.

One of the data sources, the CSV with AP information, did not originally include the attributes zone_name, parish, street and vendor. To obtain this information, two Python scripts were developed.

zone name, parish and street were obtained using the Nomatin [6] API and added to the original data source. This API returns information about the location of a set of coordinates, including the attributes mentioned above.

To retrieve AP vendor information, Wireshark [7] OUI database was used by matching the OUI part of the MAC address of the APs to public information from the database, which contains information on multiple manufacturers, containing their organizationally unique identifier(OUI).

4.6 ETL process

With the dimensional modeling complete and the data sources prepared, the following step was the implementation of the ETL process. As mentioned in Chapter 3, Python was used to perform the extract, transform, load and the Airflow tool to orchestrate the process. It is important to mention that the orchestration was done with the help of a colleague in PortoDigital, hence, the

implementation of the orchestration is not presented, only briefly described.

Given the large quantity of data stored in both databases and the necessity of performing transformations, a full load was not possible. As such, the ETL process was performed incrementally in batches, each with a month of data from the databases, orchestrated by Airflow. As for the CSV's, a full load was performed as the size of the data was much smaller.

The implementation of each step of the ETL process, namely extraction, transformation and loading are described in the following subsections.

4.6.1 Extraction

For the extraction process, the CSV's were extracted and stored in a pandas Dataframe using the Pandas library, a high-performance data analysis tool for Python, functioning as a staging area.

As for the databases, a connection was established using the library psycopg2 and the data was extracted with SQL queries and loaded to a pandas Dataframe for further transformation.

An excerpt of the script `extract_sessions` is displayed below, which connects to the databases containing FreeRADIUS session data to extract and store a month of historical data for further transformation:

```
while attempts < MAX_ATTEMPTS :
    print('_'*100)
    start = time.time()

    try :
        connection, cursor = create_connection()
        cursor.execute(
            """
            SELECT acctsessionid, acctuniqueid, username,
            nasipaddress,
                acctstarttime, acctupdatetime,
                acctstoptime, acctsessiontime,
                acctinputoctets, acctoutputoctets,
                calledstationid, callingstationid
            FROM radacct
            WHERE EXTRACT(year from acctstarttime) = %s
                AND EXTRACT(month from acctstarttime) = %s
                AND EXTRACT(day from acctstarttime) = %s;
            """,
            (year, month, day)
            )
```

```
            data = cursor.fetchall()
            break
        except Exception as e :
            print(f'Attempt {attempts + 1} of {MAX_ATTEMPTS}
            failed')
            print(e)
            attempts += 1
        finally :
            connection.close()
            cursor.close()
```

4.6.2 Transformation

With the data loaded to the staging area, the next step performed was the transformation step, to transform the data according to the dimensional model. The Pandas library, mentioned in the previous subsection, was used for the transformations.

For the dimension tables **Network Access Servers**, **Access Points** and **Domains**, the transformations performed were the simplest, as the original format from the data sources was close to the target dimension. The operations performed here were name and type changing, key generation, acquisition of foreign keys, column filtering and handling of mission values.

The **Datetime**, **Days** and **Months** tables were generated by a Python script, already in the necessary format for the dimensional model, as such no transformations were needed apart from key generation, handling missing values and acquisition of foreign keys.

The transformations applied to the raw **Wi-Fi session records** from the databases were by far the largest, since many small data quality issues observed in Chapter 3 had to be fixed in this transformations Fig. 4.2 describes the algorithm for the transformation of the Sessions fact table measures.

The key step is to identify the existence of the stop time of a session. If it does not exist, it is set to either the update time or the start time of a session if the update time is also incorrect. This may happen to the historical altered sessions mentioned in Chapter 3. If it does exist, it may also be incorrect and this needs to be verified.

Since the stop time is being modified, the total time of the session also has to be calculated again based on the stop time attributed and converted to minutes. Lastly, upload and download are converted to Megabytes

The final transformation step is the acquisition of foreign keys for the remaining attributes.

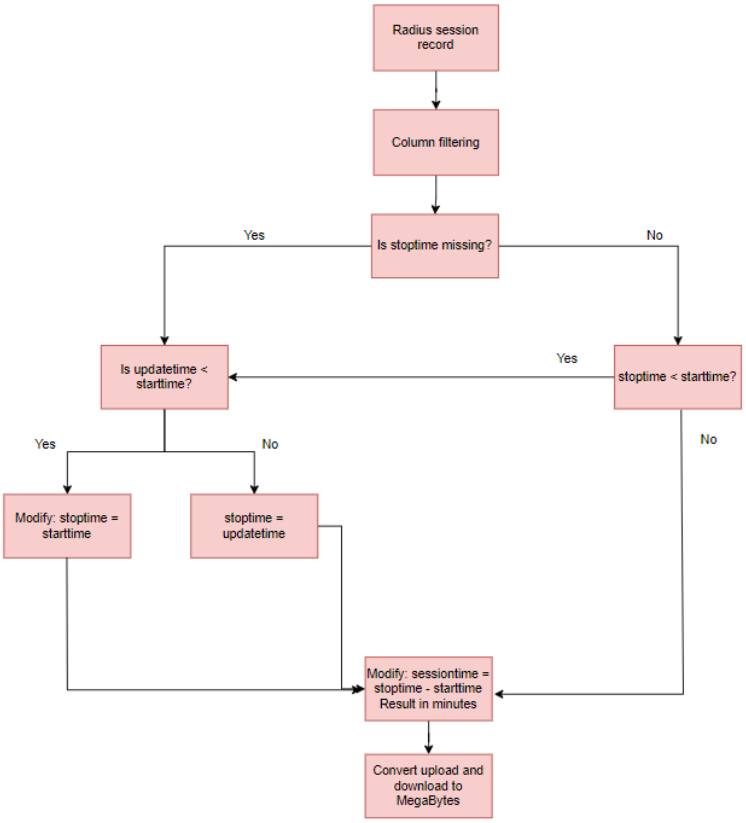


Figure 4.2: Transformation steps for the Sessions fact table measures

For the acquisition of foreign keys for the sessions fact table, there are three particular foreign keys that require additional steps: the SSID foreign key, the Domain foreign key and the Access Point foreign key to search for them, the relevant attribute needs to be extracted from the raw data.

Fig. 4.3 shows the steps for the acquisition of the foreign key for the SSID. The steps are different depending on the database used as a source, as the description is not present on the MariaDB, as such the resulting SSID extraction method is different between the two databases.

If the data source is the MariaDB, the python script for transformation of the sessions first checks for the existence of *"@"*. If it exists, it is an Eduroam session, if not, it is a PortoFreeWifi session.

For the PostgreSQL data source, the python script checks for the existence of a description and parses it to obtain the respective SSID, and if there is no description it is considered a PortoFreeWifi session.

For the acquisition of the foreign key of the Domain dimension table (Eduroam sessions only), the content of the username after the *"@"* is extracted. Finally, for the Access Points foreign key, the description is removed from the attribute called_station_id (the AP MAC address), and normalized to lowercase, so it can be searched in the Access Points dimension.

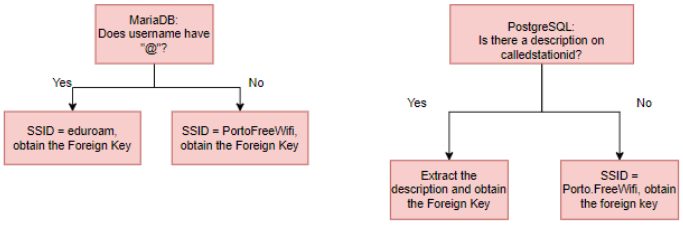


Figure 4.3: SSID Foreign Key acquisition for the sessions fact table

As there is no data source apart from the Wi-Fi session records for the **SSID's** dimension, transformation and loading are combined for this dimension, and the data is transformed according to the steps shown previously in Fig. 4.3 , extracted from the called_station_id, except there is no foreign key acquisition, only loading if it does not already exist.

The final dimension, the **Devices** dimension, also obtains data from the raw Wi-Fi session records and combines transformation and loading in the same step. The transformations applied here are described in detail for the username and user MAC address, as privacy is an important concern with sensitive data.

Most of the related work does not seem to fully provide strong privacy guarantees with MAC address anonymization, limited to hashing the MAC address, which does not provide a strong privacy guarantee. A hash on the MAC address as shown to be reversible in a matter of minutes by Demir et al. [12]. To the knowledge available during the writing of this dissertation, no solution provides strong privacy guarantees while having no loss on the information kept.

MAC address is a unique identifier assigned to the network interface controller(NIC), consisting of a 48-bit address space, with the first 3 octets corresponding to the organizationally unique identifier, meaning the manufacturer of the device, and the last 3 octets to the NIC, uniquely identifying the device, making it a sensitive piece of data.

Prior works have considered truncating the OUI to guarantee anonymization, but this has been shown to not provide strong privacy guarantees, as only 0.1% of the total OUI space has been allocated, hence the search space is, in reality, much smaller.

Our solution truncates the last hexadecimal of the MAC address and hashes it with a seed-based hashing algorithm, keeping the hash separate from the data warehouse and operational database, using it only during the data transformation process. Truncating the last hexadecimal gives a possibility of 2^4 different MAC addresses, making it impossible to reverse to the original without the use of additional information.

For the username where the sessions are Porto.FreeWifi, since the username is the user's MAC address, the same transformation is applied. The username for Eduroam does not suffer from the

same limitations of a small search space as it is in the format *institutional_username@domain*, hence, a random seed hashing with the same method as the MAC address, except the truncation part, is applied to the institutional username part.

The final transformation for this dimension is foreign key acquisition and loading for the *first_time_logged_date* attribute, requiring to search for the existence of the anonymized MAC address in the Devices table. If it does not exist, it is loaded as a new record with the date of the first time logged equal to the session start time.

4.6.3 Loading

The final stage of the ETL process is the loading step. In this step, once again Python scripts were created for loading the transformed data stored in the staging area into the postgresql DW, using the psycopg2 library for the connection.

With data transformed and ready to be loaded to the data warehouse, the first loading performed is for the Months dimension, followed by the Days and finally Datetime. These tables are the first populated in this order as most of the other tables have a need for their foreign keys.

After the dates are loaded, the next dimensions loaded are the Access Points, Domains and Network Access Servers.

The last dimensions loaded are the SSID's, Devices dimensions, and finally the Sessions fact table. The three are loaded incrementally, as mentioned at the beginning of this section, in one-month batches.

There are two stages for the entire ETL process: the historical data ETL and the the daily loading of new sessions to maintain the DW up-to-date.

The scripts used are essentially the same for the two processes but the ammount of data extracted, transformed and loaded to the DW is different. To process historical data, the ETL process extracts and integrates one month of sessions at a time. After the historical data was completely processed, the ETL process runs daily, extracting sessions from the previous day, ensuring that the DW is kept up-to-date.

Chapter 5

Results and Analysis

As stated in Chapter 1, one of the goals of this dissertation is to perform data analysis on the collected Wi-Fi data to identify patterns of the network usage by connected users, namely activity patterns, population extrapolation and user's mobility. This will, hopefully, provide a better understanding of the usage of network and the city. To perform the analysis, data from the 1st of April to the 12th of May was used, a total of 6 weeks of data, taken from the Data Warehouse. This period was chosen after most of the solutions presented in 3 were implemented and the data quality mostly stabilized.

The visualizations being presented were produced using the following python libraries: Matplotlib [15], Seaborn [31], Folium [25] and Pydeck [2] were used. All the basemaps of the city of Porto used originate from Carto [1] The methods used for the data analysis takes into account related work and suggested methods as well as the differences in the collected data. They should allow for the city policy makers to grasp the users' behaviour in terms of activity patterns and mobility within the city.

5.1 Hotspot identification

In order to facilitate the comprehension of the usage of the network in terms of the distribution of devices across the network and city hotspots, total unique devices connected per AP for the entire period were calculated and displayed over a map of the city of Porto in the form of a scatterplot, displayed in Fig. 5.1.

A clear device concentration in the downtown area can be identified and is displayed in higher detail in Fig. 5.2. It is important to mention that the downtown area has many tourist attractive areas, restaurants, and bars and is overall a very active area, so the results are within expectations.

A smaller but substantial concentration of devices can also be seen near the beach area of Porto, to the left of the city map. There is overlap between APs as they may belong to the same

hotspot, but provide connection to different areas of the street. The scatter plot produced is interactive and allows to display of information about particular APs by hovering above them.

During the study period, a total of 314715 devices connected to the network, a substantial amount, although they are not necessarily unique as explained in the limitations section in the following chapter. As such, the scatter plots may provide a good proxy to identifying areas of high concentrations of people and areas of interest in the city, specially where the density of AP's is highest, namely the downtown area.

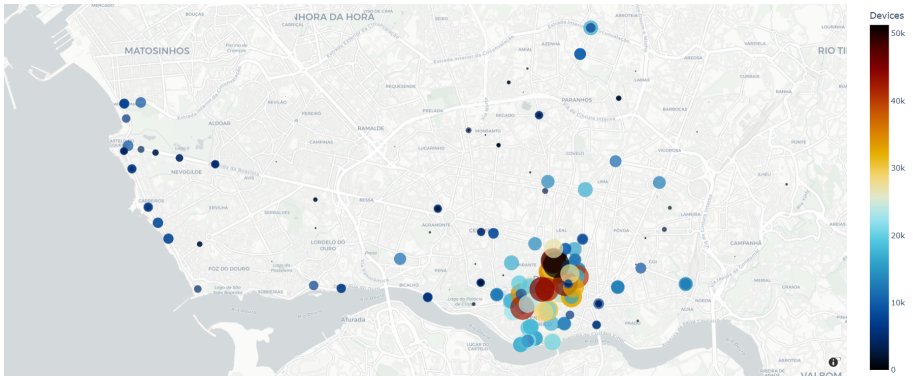


Figure 5.1: Total device counts per AP from April 1st to March 12th

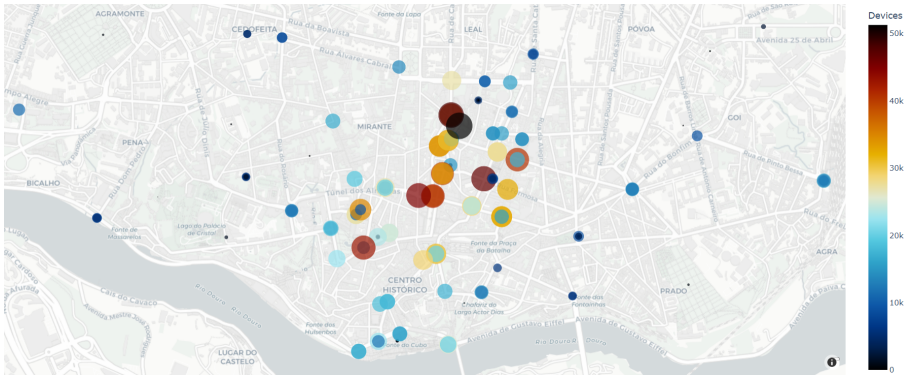


Figure 5.2: Amplified view of the downtown area

5.2 Activity patterns and population extrapolation

To extract temporal behavioural patterns of devices, the following indicators are produced to calculate the network activity baseline and standard deviation:

- *mean hourly activity counts* are calculated by:

$$\bar{U}_{n,l,t,d} = \frac{1}{N}\sum_{n=1}^{N} U_{l,t,d}$$

where U is the **activity** (number) of devices at location l, time t (hour) and day d. N is 24, the number of daily hours.

- *standard deviation* is calculated by:

$$\sigma(U_{n,l,t,d}) = \sqrt{\frac{\sum_{n=1}^{N}(U_{l,t,d} - \bar{U}_{l,t,d})}{N}}$$

The baseline obtained, illustrated in Fig. 5.3 shows a surprising regularity. For working days, the activity follows the same pattern, having a total of 3 peaks in activity at around 8 AM, 1 PM and peaks with a maximum around 4 to 5 PM. After 4-5 PM, the activity levels steadily decrease until it reaches it's lowest at around 3 AM.

As for the weekends, a pattern is also found, but it is very different from the working days period, which is to be expected. Activity counts still rapidly increases during the morning, but it is not as exponential as during week days, and it only begins to slowly rise at around 10-11 AM, much later than during when compared to working days. Activity during the afternoon is also more stable than during working days. During the late night period, activity counts are also higher than during the weekend.

Another interesting pattern is the small spike in activity during a decreasing period, specifically at Friday and Saturday night, at around 9-10 PM, and activity stays higher during the night compared to other days, This may be indicative of activity in restaurants and bars in the downtown area. Furthermore, activity levels are much higher during working days compared to the weekend, specially on Sundays where the activity counts are the lowest.

To explore the changes in standard deviation and have a more detailed perception of counts on each unique day, raw activity counts for the entire period, are calculated and displayed in Fig. 5.4. The period of 8th of April to the 24th clearly show higher activity counts than the other weeks. A sharp decrease in activity can also be seen in the 1st of May, a holiday, which also corresponds to the day with the largest standard deviation when looking at the previous figure on Wednesday, and displays a pattern similar to the weekends. This visualization may be helpful in identifying trends in population counts for a given period.

Next, to identify the existence of different user populations in the network, a metric, **regency**, is produced and defined as the time in days that a device used the city network. It is calculated as the difference in days between the start time of the first record and the stop time of the last record per device during the study period.

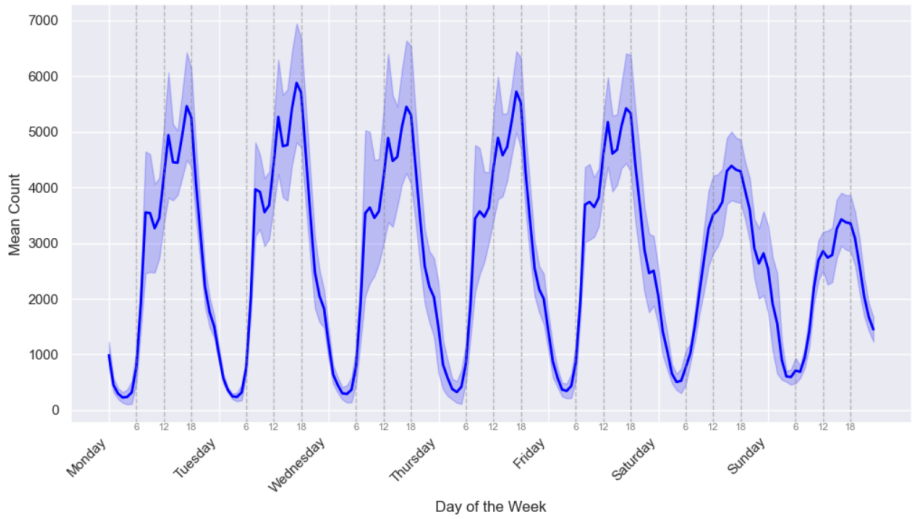


Figure 5.3: Mean activity counts and standard deviation per hour and day of the week

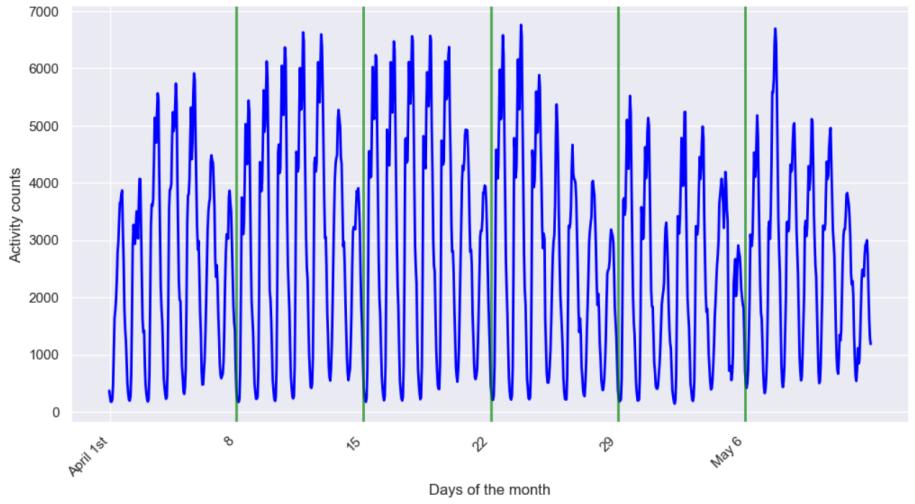


Figure 5.4: Activity counts per hour and day of the week for the period in analysis

The distribution of devices regency in days is shown in Fig. 5.5 in the form of a bar plot. More than half of the devices use the network for a period of less than 5 days and do not appear again for the duration of the study period. A clear continuous usage of the network can also be seen, as there is a large portion of devices using the network for several days and even the entire period. An important thing to note is the 0 days stay times, resulting from the pre-processing when curating stop times described in Chapter 4 for users that only connected once. This also means that a large portion of users only connect once to the network and do not use it again, at least during the study period.

From analyzing regency, the presence of different populations using the network is clear. One population continuously uses the network for a large number of days, indicative of staying in the city for large periods of time, akin to a resident or worker population. A large number of devices only utilize the network for a small period of time and are not seen again, indicative of staying in the city for a short period of time, akin to a visiting population.

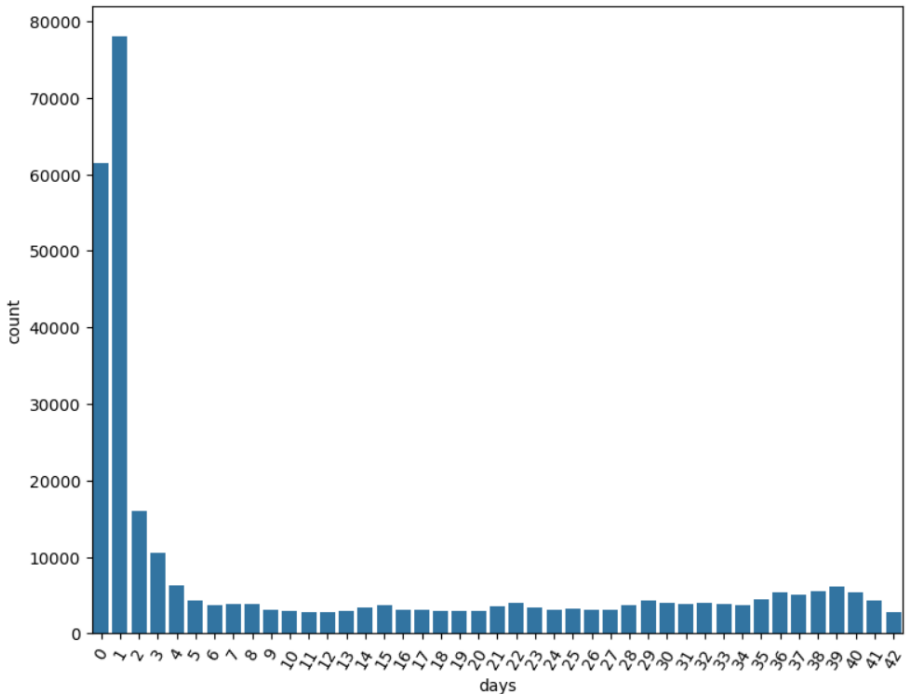


Figure 5.5: Devices regency in the network in days

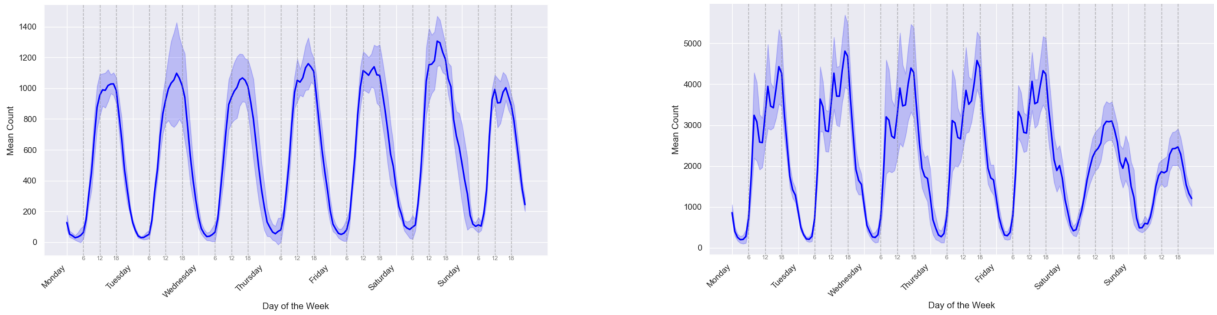
Finally, in order to differentiate and affirm differences in user populations, network activity baseline and standard deviation are calculated for two groups:

- Devices with a regency of 8 days or less
- Devices with regency of 9 days or more

The network activity baseline obtained for the two groups is displayed in Fig. 5.6. As expected, devices with higher regency in the network display very regular patterns. The 3 peaks in activity previously identified can be observed for the working days, and the peak in activity around 8 AM is even more noticeable. Devices are much more active during the working days than on weekends and the pattern drastically changes from working days to weekends. The results, taking into account the regency, clearly point to a worker or resident population.

As for the group with a regency of 8 days or less, the patterns are completely different. The activity peaks at during week days previously identified are not present, and the users seem to be more active latter in the morning/early afternoon periods, and the difference between working days and weekends seems to have a much lower impact on the network activity baseline for this group. When taking into consideration the regency of this group, it seems more indicative of a visitor profile.

This shows the usefulness of the network for both citizens of the city and visitors, as there is a considerable amount of both using the network. It can also be very useful for identifying



(a) Regency of 8 or less days

(b) Regency of 9 or more days

Figure 5.6: Mean activity counts and standard deviation for regency groups

places of interest for either population.

5.3 Mobility Analysis

To analyze the mobility of network users within the city, a *trajectory* is defined in the following format:

$$T =< P_1, P_2, ...P_n >$$

where $P_n = ID_n, X_n, Y_n, T_n$ is the nth trajectory point; ID_n is the identifier of the trajectory, (X_n, Y_n) is the location coordinates of Pn T_n is the timestamp of the location. It is important to mention that coordinates for different APs may overlap, as they are placed in the same location but provide service to different parts of the street, as such, the locations here are considered *hotspots*, the term attributed to a group of APs in the same location.

Two types of trajectories are calculated, according to the ID used for a trajectory:

- **Local trajectories** - Defined by the *session_id* and date of each session, resulting in the *hotspots* a device traversed during a single freeRADIUS session on a given day.
- **Daily trajectories** - Defined by the anonymized user MAC address and the date, resulting in the *hotspots* a device traversed during the period of a day.

The data from NAS B, representing most APs near the beach area, were excluded for this analysis, since it has not yet been standardized for the period of the collected data, resulting in one *session_id* being attributed to more than one device in a short period of time. Although the impact on data quality is overall low, for the following analysis of mobility the impact is significant, hence it was excluded.

After preprocessing, a total of 282135 local trajectories were obtained, with an average of 2,756 *hotspots* traversed per trajectory. As for daily, a total of 400757 trajectories were obtained,

with an average of 5.680 *hotspots* per trajectory.

The trajectories obtained are then converted to a network graph where for each ID, an edge is obtained if a device traverses from location A to B if $P_{n[1]}$ directly follows P_n. As such, per trajectory, records where a device remains stationary are removed and only consecutive records are considered to obtain an edge.

The nodes of the graph correspond to *hotspots* and the weight of each edge is the total number of devices that went from A to B during their trajectories according to the logic mentioned previously, equating to the flows between the two *hotspots*. Naturally, only trajectories with more than one location are considered.

For a given period, two metrics are defined to quantify movement in the city:

- Average number of *hotspots* visited per trajectory
- Total number of edges for the period.

After conversion to a network graph, a total of 1293 edges between were identified for local trajectories. For daily trajectories, a total of 3932 edges were identified. The resulting network graphs are then plotted on top of a map of the city of Porto for visualization and analysis and can be visualized in Fig. 5.7 and Fig. 5.8 respectively. Green color represents the origin of the flow and black represents the destination of the flow. The map is fully interactive and allows for a change of perspective and to zoom in and out. Hovering above the lines shows information about the flow, such as the total times the edge was traversed, the origin *hotspot* and destination *hotspot*. Line thickness also increases according to the weight attributed to each edge. In order to avoid cluttering and to display the most relevant edges, for daily flows, only edges with traffic counts above a thousand were displayed, resulting in 323 edges. As for local flows, only traffic counts above one hundred are displayed, a total of 265 edges.

For a better reference, Fig. 5.9 displays the APs that were included for the mobility analysis as to have a better indicator on which areas of the city the users of the network move between the most.

Unsurprisingly, due to the density of APs and activity counts shown in the previous section, the downtown area displays a very interconnected network graph for both types of trajectories, with the most mobility occurring at the very center of the city.

Two popular flows with a destination outside of the center can also be identified for daily trajectories, one along the river and the other to Paranhos, near an university hub, and both

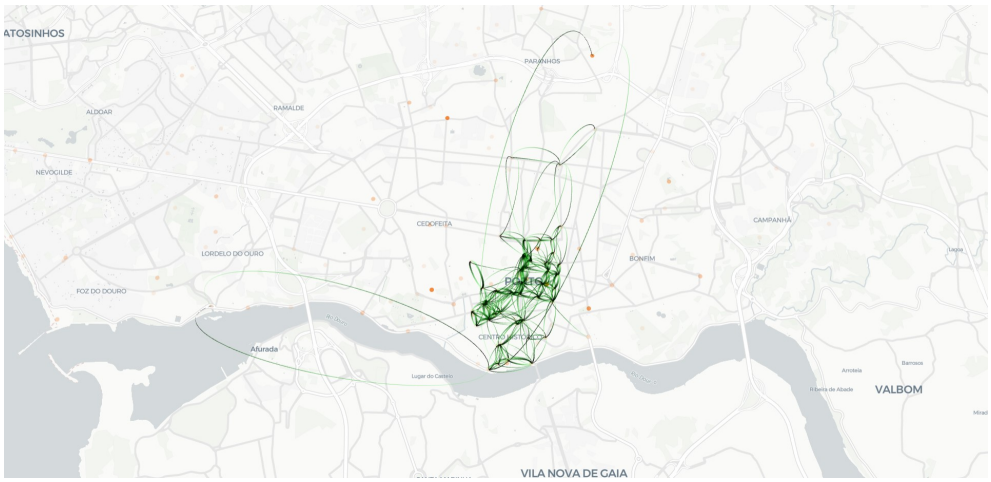


Figure 5.7: Daily flows for the whole period

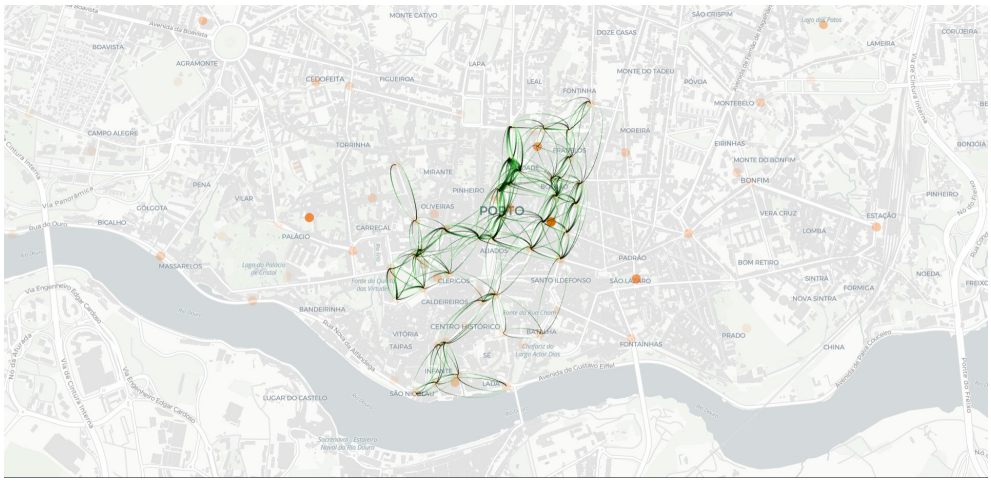


Figure 5.8: Local flows for the whole period

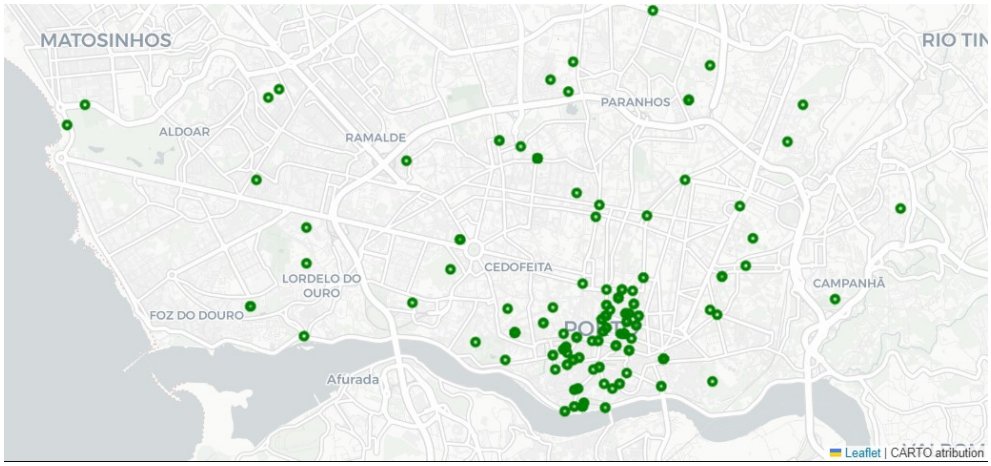


Figure 5.9: APs included for mobility analysis

flows originate or have as destination the same hotspot near the river in Rua do Infante Dom Henrique. In the downtown area, the most active locations are near the center, and the area near the river displays less movement, as visualized by the density and size of the flows. This can be best visualized in the local flows.

A significant difference in popularity in flows is clearly displayed, as the center of the city shows much more mobility compared to the area near the river, specially near Trindade, a central metro station, and Mercado do Bolhão.

This visualization may be very helpful in identifying popular mobility destinations and hubs in the city, as well as locations where an AP may be ideally placed next to better provide service to the city.

Lastly, in order to analyze differences in mobility in different periods of the day, local flows are calculated for 4 different periods:

- **Early Morning** - From 0AM to 6AM, Working Days (WD)
- **Morning** - From 6AM to 12PM WD
- **Afternoon** - From 12PM to 6PM WD
- **Night** - From 6PM to 12AM WD

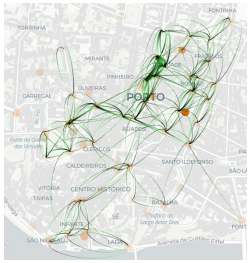
The local flows for the periods defined are calculated for the two user populations identified previously and can be visualized in Fig. 5.10. For privacy concerns, only edges with more than 10 counts are displayed. Table 5.1 displays the respective average *hotspot* traversed per trajectory and total edges for each period and user types.

For the population group with a regency of 9 or more days, mobility seems to be very dispersed around the downtown area, displaying network graphs with increasing interconnectedness of APs along the day as verified by the number of total edges, but the mobility does not seem to increase much in intensity, which can be visualized by the size of the flows, meaning that users move between various different locations but are not constantly nor are there a lot of users moving. For this population group, mobility stays active at all periods of the day.

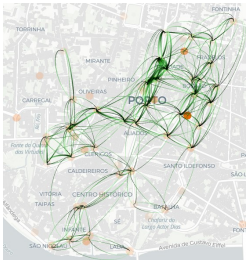
The population group with a regency of 8 days or less paints a completely different picture. Mobility during early morning is much scarcer compared to the other population group, and there is few interconnectedness between APs, as if there are a preferred set of routes for this group. Mobility starts to get more intensive during the Morning period and steadily rises until it reaches it's peak at Night where it is at it's highest but despite this the movements do not seem to be very dispersed.

An interesting observation is the intensity of mobility compared to the previous group. This group, despite having lower activity counts (shown in the previous section) and a lower number of total edges, displays a large number of counts per flows, visualized by the sizes of the flows,

at all periods except early morning, but the difference is abundantly clear for the night period. This indicates that this group is much more mobile during the day than the group with regency of 9 days or more. This visualization can be very useful to identify popular routes for visitors and areas of attraction.



(a) 0-6 AM WD period, 9 or more days regency



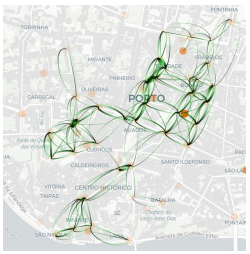
(b) 6-12 PM WD period, 9 or more days regency

(c) 12-6 PM WD period, 9 or more days regency

(d) 6-12 AM WD period, 9 or more days regency

(e) 0PM-6 AM WD period, 8 or less days regency

(f) 6AM-12 PM WD period, 8 or less days regency



(g) 12PM-6 PM WD period, 8 or less days regency

(h) 6PM - 12 AM WD period, 8 or less days regency

Figure 5.10: Week days local flows for user groups in different periods of the day

Regency	Period	Avg. *hotspots*	Total edges
>= 9 days	0AM - 6AM	2,775	266
>= 9 days	6AM - 12PM	2,678	302
>= 9 days	12PM - 6PM	2,680	345
>= 9 days	6PM - 12AM	2,741	365
<= 8 days	0AM - 6PM	2,843	112
<= 8 days	6AM - 12PM	2,674	200
<= 8 days	12PM - 6PM	2,565	213
<= 8 days	6PM - 12AM	2,718	232

Table 5.1: Average *hotspots* per trajectory and total edges for each period

Chapter 6

Conclusions

This dissertation had as the main objectives the improvement of the data collection system and the design of a BI system for Wi-Fi data in order integrate the various data sources to facilitate and speed up its analysis. Furthermore, it aimed at analyzing Wi-Fi data in order to extract meaningful information for the organization and the Municipality.

In order to achieve the objectives, a study on BI concepts and state-of-the-art as well as a review on studies performed for urban analysis using Wi-Fi data to extract useful applications and methods for the analysis.

Before the design of the architecture, an in depth study of the FreeRADIUS tool and analysis on the data quality and the data collection system was performed to not only provide solutions to the problems described, but also to further identify the requirements the architecture would need to support a real-time operational data source. From this, several data quality problems were identified and corrected, and the data collection system was standardized to minimize future errors. This was undoubtedly the most challenging and time consuming task, as there was almost no previous knowledge of the components involved in the collection system or the tool FreeRADIUS tool so getting to the root problem was a arduous challenge.

Then, with the requirements fully defined, the architecture, following a bottom-up approach with dimensional modeling, was designed and implemented. To migrate the large amounts of historical data and integrate the various data sources in the Data Warehouse as well as to keep the DW up-to-date on a daily basis, an ETL process was implemented using Python for the code and Airflow for orchestrating the ETL process. During the transformation process, privacy is also taken into consideration and the solution provided guarantees that the original MAC address can not be retrieved. Furthermore, materialized views were created to optimize query efficiency for reports and dashboards. Currently, two dashboards already make use of the Data Warehouse implemented, for reporting.

Lastly, analysis on 6 weeks of data, selected after the data quality mostly stabilized was performed, with a total of 314715 devices. For the analysis, methods for hotspot identification, activity patterns and population extrapolation and finally mobility were applied and several

interactive visualizations were produced such that a sense of the city and meaningful information can be extracted through the visualizations. Behaviour was studied trough activity patterns, where very regular patterns in activity were detected. Two types of user populations were extrapolated: a residential/worker population and a visitor population, demonstrating the usefulness of the network for the City of Porto. In terms of mobility, a network graph approach was selected to visualize the flows in the city. This approach allowed for the visualization of flows in a map of the city of Porto in an interactive way where mobility intensive streets can be easily identified.

Overall, the objectives defined for this dissertation were achieved. The difficulties faced during this project working with different technologies and areas of expertise served as an enriching experience, both on a personal and professional level.

6.1 Limitations

Works based on Wi-Fi data suffer from several limitations. The number of devices does not directly equal number of users, as a single user may carry more than one device. Another limitation is that when users change SSID, if they have privacy protecting enabled devices on MAC address which currently is the default option in most Apple devices, the device changes MAC address for the new SSID. This means that the same user may be counted more than once even if it is the same device.

Another limitation regarding the number of users is the approach taken for privacy protection of pruning the last MAC address, which introduces collision between devices, resulting in an impact in the total number of devices. Finally, since the data utilized originates from Wi-Fi session logs, naturally only users using the network were captured, meaning the results obtained represent the population of users in the city that use the network but not the entire population of the city of Porto.

The last limitation is related to the data excluded for the analysis of mobility. As mentioned previously, unfortunately data from the NAS containing most APs near the beach area was not included as it was not standardized in time for the analysis due to technical difficulties with this particular NAS, resulting in an impact on the analysis of mobility for this area of the city.

6.2 Future Work

Although the objectives for the dissertation were achieved, there are still improvements that can be made to the analysis, such as exploring different applications for the Wi-Fi data that may be useful to the city, as well as expanding the period of analysis. An application can also be developed to explore the visualizations created with more freedom to select periods or specific days for mobility analysis, for example.

Another possible route is the development of a real-time Operational Data Store, with limited historical data, for visualization of Wi-Fi in the city in real-time.

Bibliography

[1] Basemaps | CARTO — carto.com. https://carto.com/basemaps. [Accessed 01-07-2024].

[2] Gallery &x2014; pydeck 0.9.1 documentation — deckgl.readthedocs.io. https://deckgl.readthedocs.io/en/latest/#. [Accessed 01-07-2024].

[3] FreeRADIUS — freeradius.org. https://freeradius.org/. [Accessed 30-06-2024].

[4] How Does RADIUS Authentication Work? — networkradius.com. https://networkradius.com/articles/2019/05/08/how-does-radius-authentication-work.html, . [Accessed 30-06-2024].

[5] How does RADIUS Accounting work? — networkradius.com. https://networkradius.com/articles/2019/06/05/how-does-radius-accounting-work.html, . [Accessed 30-06-2024].

[6] Nominatim Manual — nominatim.org. https://nominatim.org/release-docs/develop/. [Accessed 30-06-2024].

[7] Wireshark · Go Deep — wireshark.org. https://www.wireshark.org/. [Accessed 30-06-2024].

[8] Martin Aruldoss, Miranda Lakshmi Travis, and V Prasanna Venkatesan. A survey on recent research in business intelligence. *Journal of Enterprise Information Management*, 27(6): 831–866, 2014.

[9] Pierfrancesco Bellini, Daniele Cenni, Paolo Nesi, and Irene Paoli. Wi-fi based city users' behaviour analysis for smart city. *Journal of Visual Languages & Computing*, 42:31–45, 2017.

[10] Surajit Chaudhuri and Umeshwar Dayal. An overview of data warehousing and olap technology. *ACM Sigmod record*, 26(1):65–74, 1997.

[11] Timothy Chee, Lee-Kwun Chan, Min-Hooi Chuah, Chee-Sok Tan, Siew-Fan Wong, William Yeoh, et al. Business intelligence systems: state-of-the-art review and contemporary applications. In *Symposium on progress in information & communication technology*, volume 2, pages 16–30, 2009.

[12] Levent Demir, Mathieu Cunche, and Cédric Lauradoux. Analysing the privacy policies of wi-fi trackers. In *Proceedings of the 2014 workshop on physical analytics*, pages 39–44, 2014.

[13] Sophian Gauzelin and Hugo Bentz. An examination of the impact of business intelligence systems on organizational decision making and performance: The case of france. *Journal of Intelligence Studies in Business*, 7(2), 2017.

[14] Marcus Handte, Muhammad Umer Iqbal, Stephan Wagner, Wolfgang Apolinarski, Pedro José Marrón, Eva Maria Muñoz Navarro, Santiago Martinez, Sara Izquierdo Barthelemy, and Mario González Fernández. Crowd density estimation for public transport vehicles. In *EDBT/ICDT Workshops*, pages 315–322. Citeseer, 2014.

[15] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi:10.1109/MCSE.2007.55.

[16] William H Inmon. *Building the data warehouse*. John wiley & sons, 2005.

[17] Ralph Kimball and Margy Ross. *The data warehouse toolkit: the complete guide to dimensional modeling*. John Wiley & Sons, 2011.

[18] Constantine E Kontokosta and Nicholas Johnson. Urban phenology: Toward a real-time census of the city using wi-fi data. *Computers, Environment and Urban Systems*, 64:144–153, 2017.

[19] Nilesh Mali and Sachin Bojewar. A survey of etl tools. *International Journal of Computer Techniques*, 2(5):20–27, 2015.

[20] Rajendrani Mukherjee and Pragma Kar. A comparative review of data warehousing etl tools with new trends and industry insight. In *2017 IEEE 7th International Advance Computing Conference (IACC)*, pages 943–948. IEEE, 2017.

[21] Solomon Negash and Paul Gray. Business intelligence. *Handbook on decision support systems 2*, pages 175–193, 2008.

[22] Muhammad Obeidat, Max North, Ronny Richardson, and Vebol Rattanak. Business intelligence technology, applications, and trends. 2015.

[23] Larissa Oliveira, Katia Obraczka, and Abel Rodríguez. Characterizing user activity in wifi networks: University campus and urban area case studies. In *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems: Malta, Malta*, 2016.

[24] Celina M Olszak and Ewa Ziemba. Approach to building and implementing business intelligence systems. *Interdisciplinary Journal of Information, Knowledge, and Management*, 2(1):135–148, 2007.

[25] python visualization. Folium.

[26] Allan Rubens, Carl Rigney, Steve Willens, and William A. Simpson. Remote Authentication Dial In User Service (RADIUS). RFC 2865, June 2000. doi:10.17487/RFC2865.

[27] J Sreemathy, R Brindha, M Selva Nagalakshmi, N Suvekha, N Karthick Ragul, and M Praveennandha. Overview of etl tools and talend-data integration. In *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, volume 1, pages 1650–1654. IEEE, 2021.

[28] Martin W Traunmueller, Nicholas Johnson, Awais Malik, and Constantine E Kontokosta. Digital footprints: Using wifi probe and locational data to analyze human mobility trajectories in cities. *Computers, Environment and Urban Systems*, 72:4–12, 2018.

[29] Panos Vassiliadis and Alkis Simitsis. Extraction, transformation, and loading. *Encyclopedia of Database Systems*, 10:14, 2009.

[30] Carlo Vercellis. *Business intelligence: data mining and optimization for decision making*. John Wiley & Sons, 2011.

[31] Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021. doi:10.21105/joss.03021.

[32] Hugh J Watson. Recent developments in data warehousing. *Communications of the Association for Information Systems*, 8(1):1, 2002.

[33] Lamia Yessad and Aissa Labiod. Comparative study of data warehouses modeling approaches: Inmon, kimball and data vault. In *2016 International Conference on System Reliability and Science (ICSRS)*, pages 95–99. IEEE, 2016.

[34] Guilherme Augusto Zagatti, See-Kiong Ng, and Stéphane Bressan. A data warehouse of wi-fi sessions for contact tracing and outbreak investigation. In *Transactions on Large-Scale Data-and Knowledge-Centered Systems XLVIII: Special Issue In Memory of Univ. Prof. Dr. Roland Wagner*, pages 85–104. Springer, 2021.