


Article

Pervasive Real-Time Analytical Framework—A Case Study on Car Parking Monitoring

Francisca Barros ¹, Beatriz Rodrigues ¹, José Vieira ² and Filipe Portela ^{1,2,*} 
¹ Algoritmi Centre, University of Minho, 4800-058 Braga, Portugal

² IOTech, 4785-588 Trofa, Portugal; josevieira@iotech.pt

* Correspondence: cfp@dsi.uminho.pt

Abstract: Due to the amount of data emerging, it is necessary to use an online analytical processing (OLAP) framework capable of responding to the needs of industries. Processes such as drill-down, roll-up, three-dimensional analysis, and data filtering are fundamental for the perception of information. This article demonstrates the OLAP framework developed as a valuable and effective solution in decision making. To develop an OLAP framework, it was necessary to create the extract, transform and load the (ETL) process, build a data warehouse, and develop the OLAP via cube.js. Finally, it was essential to design a solution that adds more value to the organizations and presents several characteristics to support the entire data analysis process. A backend API (application programming interface) to route the data via MySQL was required, as well as a frontend and a data visualization layer. The OLAP framework was developed for the ioCity project. However, its great advantage is its versatility, which allows any industry to use it in its system. One ETL process, one data warehouse, one OLAP model, six indicators, and one OLAP framework were developed (with one frontend and one API backend). In conclusion, this article demonstrates the importance of a modular, adaptable, and scalable tool in the data analysis process and in supporting decision making.

Keywords: framework; OLAP model; ioCity; ioScience; business intelligence; data warehousing



Citation: Barros, F.; Rodrigues, B.; Vieira, J.; Portela, F. Pervasive Real-Time Analytical Framework—A Case Study on Car Parking Monitoring. *Information* **2023**, *14*, 584. <https://doi.org/10.3390/info14110584>

Academic Editor: Kostas Vergidis

Received: 26 July 2023

Revised: 16 October 2023

Accepted: 17 October 2023

Published: 25 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The current volume, variety, and velocity of data production pose significant challenges for industries. The sheer amount of data being generated requires efficient storage and management solutions. The diverse types of data (structured, semi-structured, and unstructured) complicate the extraction of meaningful insights. Additionally, the fast-paced nature of real-time data streams demands quick processing capabilities. To address these challenges, an efficient OLAP framework is essential, enabling organizations to perform multidimensional analysis, gain insights, and make informed decisions in the face of the overwhelming data landscape.

Online analytical processing (OLAP) is a technical approach that allows users to analyze relevant information from different points of view and in various combinations. By connecting to a database, the system enables the visualization of statistical reports that allow extracting, consulting, and retrieving data [1].

The end-users usually do not know how to easily create dashboards that interact with the data, or of the existence of open-source platforms that are able to do that. For example, through cube.js, an OLAP model can be derived. Cube.js is an open-source headless business intelligence (BI) that allows pre-calculating and pre-aggregating data, creating an OLAP cube that makes analysis faster. Cube.js enables the connection to several databases, and consequently, the schema is defined for each dimension and fact table in several cubes. The foreign keys and the relationships between the other cubes are limited for fact tables, and the facts are calculated. We can save the desired combinations through pre-aggregations for faster analysis [2]. To test it, a proof of concept was developed using a

research project. However, cube.js presented some disadvantages, such as the absence of drill-down and roll-up methods, limited options for types of graphs, and lack of integration with software as a service (SaaS) platforms. These analyses are essential to obtain a clarified observation of data.

Moreover, the previous research work allowed the team to understand that no tool can perform data science as a service using structured and unstructured data and access them using an offline connection. That work originated the development of a patent [3] that will be explored in this case study.

So, to address both issues, a new artifact was created. The main goal of the work presented in this paper was to build an OLAP framework that is able to include the main patent findings (offline analysis and structured and unstructured data sources) and the fundamental characteristics of the data analysis process by turning it into a modular solution that is fully scalable and adaptable to any industry.

Then, a proof of concept was produced to prove the feasibility of the developed framework. The case study had two phases, the first using only cube.js and the second using the framework developed. This split reinforces the need for a global framework such as the one set out here. Many of the limitations that led to the framework development were found in phase one. The chosen technologies follow the guidelines identified after the research was carried out within the scope of the ioScience patent [3,4].

Experiments were conducted using data collected from city sensors, specifically focusing on park movement management. Therefore, the utilization of this data indicates that this work is part of the ioCity project [5].

This research project is supported by public funds and aims to help solve city challenges such as mobility. In this way, ioCity proposes, through the analysis of a set of data (sensors, occupancy rate, traffic, parking, among others), to help the customer make a more informed and real-time decision about car parks and public transport. After analyzing various data, it is expected that a prototype will be designed and developed to showcase a decision-making support system that can provide valuable assistance to the customer. This article also describes the OLAP framework that can be integrated into ioCity.

To achieve the main objective, some secondary objectives were defined: extract, transform, and load the provided data (ETL); create a data warehouse (DW); interpret the problem and perception of reality; develop Key Performance Indicators (KPIs); find patterns among the data and draw relevant conclusions that are important to support decision making (data science); and develop an architecture that responds to the requirements.

The case study presented in this paper is an excellent example of how to turn IoT data into smart and valuable information.

This paper is organized as follows: the Introduction presents the basis for this work. The Section 2 discusses the background of the article, covering topics such as smart cities and data analytics, as well as presenting the related work. The Section 3 describes the materials and methods used. The fourth chapter discusses the developed case study, and the fifth presents the OLAP framework and the results achieved with the case study data. In the Section 6, there is a general discussion of the outcomes derived from this article. The Section 5 serves as the conclusion, presenting the final ideas and findings of the study.

2. Background

Before presenting the background, it is important to explain the basis of this work. The article presents some parts of one ongoing study. The technical part follows the patented directives [3], and the proof of concept follows the achievements of a realized study (three steps), i.e., implements the ideas identified. During the patent process, no tool was found that allows offline data science as a service and includes several data source types. Unfortunately, therefore, the technical part of the patent cannot be represented here, so it is intrinsically shown in the solution (Section 5).

Thus, in this section, the content of each piece will be briefly discussed to help understand this article. Figure 1 presents the workflow of this paper. Initially, a study of

the projects and initiatives of smart cities was developed [6]. Then, variables influencing citizens' decision making in public or private transport were identified. Through the data received by the city council, and with the identified variables, it was necessary to build a prototype with cube.js to visualize the data [7]. Finally, developing an OLAP framework complementary to the prototype was proposed, aiming to be a modular and versatile solution for the industry.

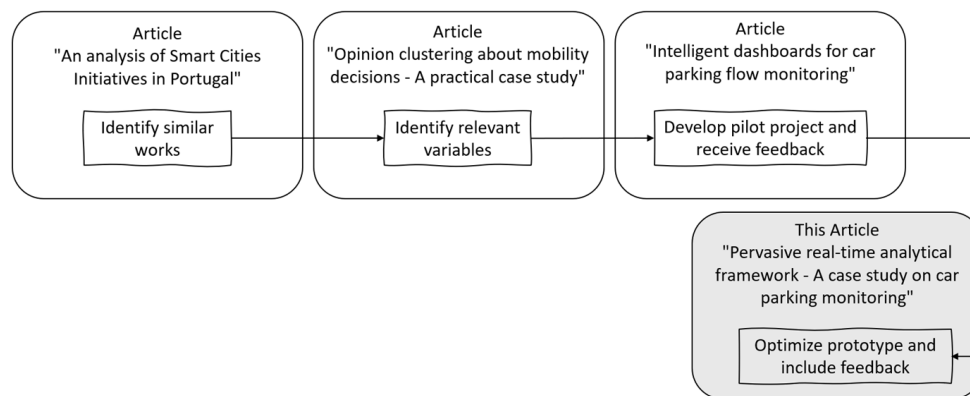


Figure 1. Workflow of the paper.

Before describing the mentioned articles, a brief explanation of some related terms is presented. Moreover, to develop this new framework, some well-known tools, such as Power BI, Tableau, and Pentaho were explored.

2.1. Smart Cities

The smart city concept emerged in the last decade and can be considered an urban area that integrates and uses a wide range of electronic methods and sensors to capture data. To create new opportunities that improve the quality of life, this concept “coordinates and integrates several technologies that hitherto been developed separately from each other, but have clear synergies in their operation” [8]. For the city to grow, a variety of data is analyzed encompassing citizens, buildings (hospitals, schools), transport and urban mobility, and use of waste, among other data, so that there is growth and city efficiency with the capacity to “integrate and add value to the provision of urban services” [8].

2.2. Data Analytics

Understanding and analyzing the data in detail are essential to get the most out of it and obtain the best indicators for the business. With this detailed analysis, it is easier to perceive the opportunities and threats of an organization (for example). For this reason, data are an asset for companies because, with them, it is possible to draw basic conclusions about competitiveness and success [9]. The reality and the context can be realized and valuable insights can be extracted from the data that help the decision-making process [10].

2.3. Extract, Transform, and Load (ETL)

The ETL process is divided into three phases: extract, transform, and load. First, the data were analyzed to understand how they were organized and what was most important for the final objective [11]. Then, in the extract phase, the data are copied or exported from source locations to a staging area. After that, in the transform phase, raw data are subject to data processing, for example, correction of the errors detected. Finally, in the load phase, the desired columns are loaded for the data warehouse dimensions. Fact tables are usually loaded through associations to dimensions and facts [12].

2.4. Key Performance Indicators (KPIs)

Key Performance Indicators (KPIs) are management tools that monitor the performance and success of an organization's objective. Through these indicators, it is possible to quantify the status of the purposes and understand the best strategy to achieve the success of the established goal [7].

2.5. Data Warehouse

Data warehouses (DWs) are central data repositories from one or more sources and can store current and historical data in a unique location. A multidimensional model is a type of modeling used in the design of data warehouses. This type of modeling emphasizes identifying, modeling, and implementing critical business processes. A multidimensional model is composed of fact and dimension tables. The fact tables contain the business indicators necessary for the intended analyses, while dimension control descriptions describe the points [13].

2.6. OLAP

Online analytical processing (OLAP) is a tool that allows users to analyze relevant information from different points of view and in various combinations. By connecting to a database, the system enables the visualization of statistical reports that allow extracting, consulting, and retrieving data [1]. This area has been extensively explored more recently. With information technology, business development has greatly impacted local, national, and international business activities. There are many frameworks of OLAP for business intelligence, which offers a range of tools and features for data integration, visualization, and analysis [14].

Some examples of how OLAP processes, such as drill-down, roll-up, and three-dimensional analysis, aid in data perception and decision making are:

- Drill-down: OLAP allows users to navigate from summarized data to detailed information. This process helps uncover specific patterns or outliers that may not be apparent at a higher level of aggregation, leading to a deeper understanding of the data.
- Roll-up: Conversely, roll-up allows users to aggregate data from a detailed level to a higher level of abstraction. Roll-up facilitates summarization, simplifies complex data, and provides a broader perspective, which can be useful for decision making and identifying overall trends and performance indicators.
- Three-dimensional analysis: OLAP presents data in a multidimensional structure, typically represented as a data cube. This helps in-depth exploration and better understanding of complex relationships within the data.

2.7. Business Intelligence Tools

As mentioned before, there are no similar tools that solve the same issues addressed by the patent. However, well-known tools are used to do OLAP tasks and were used as a basis to produce this framework, i.e., Power BI, Tableau, and Pentaho.

Power BI is an enterprise BI platform. It allows connection to data and subsequent creation of visual reports that help decision making [15]. Despite the possibility of creating reports with an appealing graphical interface and the number of tools available for creating specific graphics for each client, this tool does not allow integration into software as a service (SaaS) [16,17].

Tableau is a visual analytics platform that allows the user to perceive anomalies and extract information from data to solve problems and maximize the industry's value [18]. Although Tableau is a tool that enables creating reports and providing interactive visualizations, it does not provide the feature of automatically updating reports. In addition, it is also not a SaaS service; it has limited data pre-processing features and does not provide customized information for the client [19].

Pentaho is free software for business intelligence systems. This tool is similar to others systems such as Power BI and Tableau. A cube is created, and reports are generated through

the data warehouse connection [20]. Pentaho has five different software packages: “Pentaho BI Platform” (used to build a data warehouse and development reports); “Pentaho Data Integration” (used for the ETL process); “Mondrian” (used to create an OLAP server); “Pentaho Reporting” (used only for creating reports); and “Weka” (offers many tools for machine learning and data mining). For each one, many features are available. By using all of these software packages, better performance is achieved because they complement each other, from the connection to the data warehouse to tools for machine learning and data mining [21].

2.8. Related Works

In this section, three articles are presented, in chronological order of development, which led to the creation of the present study. Digital transformation in cities is increasingly being developed, and it must add value to the habitants. In this way, the variables that interfere with the decision regarding public transport versus individual transport were collected. As previously mentioned, this study is part of the ioCity project, which aims to solve mobility problems and help citizens access helpful information. Consequently, through the data obtained at the mobility level, the ETL process and a data warehouse were created to construct the OLAP model through the cube.js tool. Finally, a study of innovative projects in smart cities was developed to understand the current situation in Portugal.

The first article [6], “An analysis of Smart Cities Initiatives in Portugal”, is a study that aims to demonstrate the importance of technological evolution and the impact technology has on cities, and to analyze the current state of smart cities in Portugal. Information was collected on sixty-eight initiatives (sixty-two city projects and six national projects) in Portugal that are being developed or implemented. This study presents strategies and initiatives to be adopted for developing smart cities in Portugal. Through state-of-the-art development and with all the initiatives found, it is possible to identify projects that would be an asset to the ioCity project in the field of mobility.

The article “Opinion clustering about mobility decisions—A practical case study” shows that the digital transformation of cities must follow the population’s priorities to add value to the town. This article demonstrates and presents the variables that citizens prioritize in decision making between public transport choices. Three scenarios were created, and the variables (fuel cost, average park price, total capacity, occupancy percentage, and number of transfers between modes of transport) have a high degree of importance. This study was a starting point for data collection and, consequently, to respond to the need to optimize and build intelligent models that meet the demands of the population and the city. The next step is related to the development of an OLAP and predictive models [22].

“Intelligent dashboards for car parking flow monitoring” presents good practices of a business intelligence process and how an organization can succeed, as a contribution to the data warehouse, the OLAP model, and the step-by-step process that must be followed. An ETL process and a data warehouse were developed. The cube.js tool was used to create an OLAP model to store data. Schemas and pre-aggregations were defined to support the parking flow monitoring. These tasks allowed the extraction of valuable information and assisted in decision making. The speed and veracity of the indicators obtained enabled the team to conclude that the OLAP mechanism is an asset that allows decision makers to have a quality data analysis tool [6].

This visualization platform has some flaws in terms of data analysis. Thus, there is a need to develop a complete OLAP framework. This paper is an extension of the three papers outlined in this section, as it builds upon the work presented in those papers.

2.9. OLAP Applications

OLAP can be applied in all areas of knowledge. In this subsection, an example study in the field of construction is presented to showcase the application of OLAP. In civil construction projects, work delays and budget increases during the construction process are recurrent, causing problems between the owners and those responsible for the work.

This is due to inadequate management of labor resources. Thus, a detailed analysis was conducted at the level of industrial construction, and data collection was carried out for a later study and with the aim of overcoming this management problem. Consequently, a data warehouse was built for further OLAP analysis with the collected data. In this way, reports and dynamic graphs of online analytical processing were obtained, where it was possible to extract critical knowledge for better management of the civil construction industry. The results obtained make it possible to achieve more detailed planning of the work, and more accurate and effective management of resources, which reduces delays and potential changes in the value of the work [23].

The need to extract knowledge from different industries leads to developing an adaptable and scalable framework, which allows working without internet access and is a standard for any data source.

3. Materials and Methods

This section describes the materials and tools used for this paper. Design Science Research (DSR) was used for the OLAP framework demonstration, and the Kimball Methodology was used for the case study. SCRUM was used for more effective project management. Finally, all tools used are described.

3.1. Design Science Research (DSR)

The DSR methodology is essential in information systems since it is oriented toward creating artifacts. The techniques and perspectives that this methodology offers for scientific research promote the quality of functional performance of information systems [24]. Thus, there are six fundamental steps, which can be seen in Figure 2.

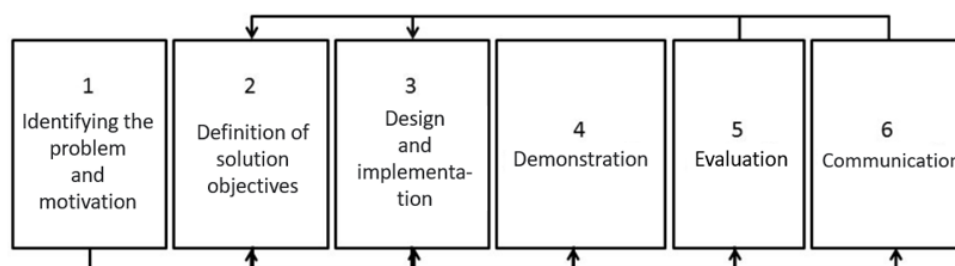


Figure 2. Design Science Research methodology, adapted from [20].

The following sections describe the DSR activities in detail [24]. All these activities were carried out within the framework of this project.

3.1.1. Identifying the Problem, Motivation, and Definition of Solution Objectives

Initially, it was necessary to identify the specific research problem. Secondly, the state of the art was surveyed so that knowledge obtained helps in the definition of the problem and helps to design a solution. Through the creation of artifacts and the knowledge acquired, it becomes possible to progress and implement solutions based on technologies to solve problems. This project arises from the need to change mobility alternatives due to excess congestion, pollution, and accidents. The problem and motivation were previously described in Section 1—Introduction.

After the development of the first phase, identification of the problem, and motivation, the objectives are defined. It is expected that there is knowledge of the state of the problem and the solutions available in the market so that the solution's purpose is innovative and efficient. This paper aims to build a modular OLAP framework that can be integrated into any industry with fundamental characteristics such as drill-down and roll-up, and the ability to create analytical dashboards.

3.1.2. Design and Implementation

In phase three, an artifact containing the solution's underlying functionalities is proposed, as well as its definition and characteristics. The prototype's architecture helped construct the actual artifact so that it was coherent. This artifact was based on models, methods, algorithms, and techniques. The entire data analysis process, ETL, DW construction, and OLAP models were developed at this stage.

3.1.3. Demonstration

This step is expected to demonstrate how the developed artifact solves the previously identified problem. For the proof of the artifact in solving the problem to be effective and valuable, it is necessary to create a well-defined and appropriate environment, which may involve experiments, simulations, and applications in case studies. All the artifacts developed are presented in Section 5.

3.1.4. Evaluation

The evaluation consists of observing and measuring how the artifact supports the solution of the problem and if it meets the requirements demanded and initially defined. The assessment can be performed using different methods depending on the situation. A comparison can be made of the features developed with those predefined in the artifact, or, for example, to quantitatively measure the performance through simulations.

Through Figure 2, where all the activities that must be performed using the DSR methodology are represented, it is possible to perceive that the process can be executed in sequence or follow a different order. In phase 5, evaluation, or the last stage, communication, there may be a setback to activity 2 or 3, depending on the evaluation results, where the objectives may have to be redefined.

3.2. Tools

For the selection of the tools to be used, an analysis was conducted, as presented in Section 2—Background, within the scope of the ioScience patent [25,26]. However, since none of the explored tools adequately addressed the problem, the decision was made to develop a new customized artifact specifically designed to solve the problem at hand. Thus, in Table 1, the tools selected for the execution of the project are presented. For example, the Python language was used for the entire data analysis and ETL process.

Table 1. Tools.

Tool Name	Type	Functionality
Microsoft Office Excel (2016)	Calculation Sheet	Support operations for data analysis.
Python (2.8.2)	Programming language	Development of the data model and prototype.
MySQL Workbench (8.0.31)	Database design tool	Creation and maintenance of data model.
Pandas Library (2.0.3)	Open-source software	Tool for data analysis and manipulation.
Past (4.0.3)	Statistical software	Paleontological Statistics is a software program for statistical data analyses.
Jupyter Notebook (8.3.0)	Open-source software	Interactive web-based computing platform where open-source code can be developed.
Cube.js (0.31.35)	Open-source software	Interactive platform for building analytical web applications.
Visual Studio Code	Open-source software	Source code editor.
Vue.js (2.7.7)	Programming language	Frontend JavaScript framework for viewing interface development.

Table 1. Cont.

Tool Name	Type	Functionality
Vuetify (2.6.14)	Graphical user interface library	A framework that helps web development, providing helpful design tools for the frontend of the OLAP framework.
Apache ECharts (6.5.1)	Open-source JavaScript preview library	Development of intuitive, interactive, and customizable visualization charts.
Google Maps (2.0.15)	Web mapping platform	Geolocation system through dynamic maps.
Node.js/Express.js (17.9.1)	Backend framework for API development	Backend API to define data routes by MySQL.

3.3. Kimball Methodology

As this is a business intelligence study, the methodology adopted was the Kimball Lifecycle. Kimball's methodology focuses on designing a data warehouse by creating fact tables that correspond to business processes. To that end, Kimball developed a method for developing these, called the Lifecycle roadmap.

The diagram represents the sequences, dependencies, and concurrency of the tasks that must be performed, enabling work development correctly within the set time.

In Figure 3, Kimball's scheme that summarizes the methodology is presented [25].

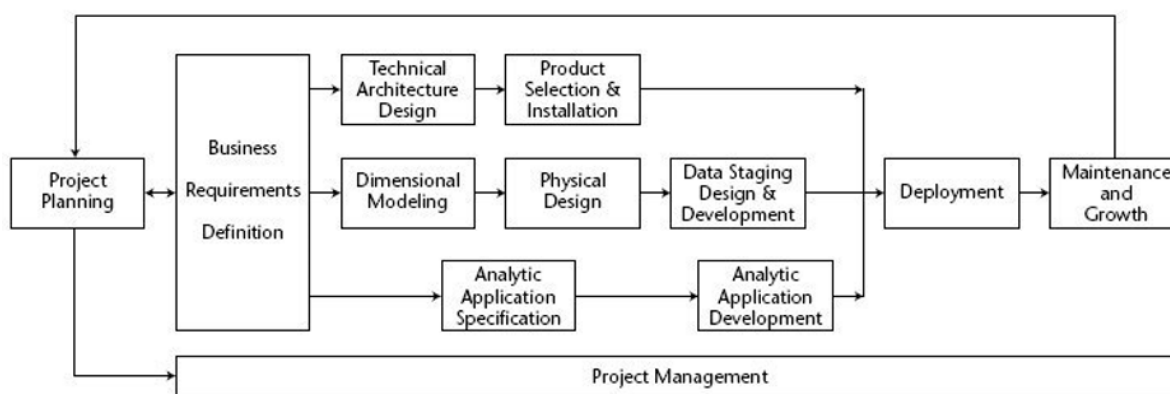


Figure 3. Kimball Methodology.

The Kimball Lifecycle methodology includes several fundamental tasks that were carried out in the context of the project, namely [25]:

- Project Planning: Planning phase (defined scope and planned project).
- Project Management: This phase extended throughout the project to manage the team, resources, and expectations.
- Business Requirements Definition: The business requirements compression phase had the fundamental objective of linking activities with the lifecycle.
- Technology Track—This stage contained two milestones:
 - Technical Architecture Design: Consists of developing the architecture of the business;
 - Product Selection and Installation: Phase in which the architecture is used to complete data warehousing.
- Data Track—This phase contains three milestones:
 - Dimensional Modeling: Consists of designing a multidimensional model based on business requirements;
 - Physical Design: Proceeds to the design of the database;
 - Data Staging, Design, and Development: Extracts, adapts, and provides data and presentation and management of the ETL system.

- Business Intelligence Track—This stage contained two milestones:
 - Analytic Application Specification: Phase in which applications are selected to support business requirements;
 - Analytic Application Development: Uses design to develop and validate applications that support business requirements.
- Deployment: In the case of data warehousing/business deployment intelligence, data are the primary input. Processing the data in ETL is the most unpredictable task. Unfortunately, even if the data are not fully prepared, implementation proceeds. This phase required pre-deployment testing, documentation, training, and maintenance and support.
- Maintenance and Growth: After deployment, maintenance must be carried out to control the system's satisfactory performance. If the project needs to grow because of new data or objectives, the lifecycle starts again.

3.4. SCRUM

The SCRUM methodology was used for project management to improve project efficiency and reduce wasted time and resources. First, it was necessary to define and understand the project's role, then list the priorities, define, and plan the sprints, and, later, the weekly and daily meetings [26].

Sprints are a sequence of activities with a defined deadline, which must be developed during that time. In this way, there is a fortnightly meeting to determine the tasks for the sprint (sprint planning) where the project owner, based on the product backlog (list of all desired product features), divides the tasks for the sprint in question. During the two weeks of work, through daily meetings, each project member shares the activities to be carried out for the day, what he did the day before, and respective difficulties. After the end of the sprint, there is a meeting to validate all the work developed and a new session for sprint planning [26]. The fact that the team holds all the meetings and plans for the entire project ensures that the final product is high quality.

Kimball's methodology arises as a response to the problem encountered in Section 3.1.1 of the DSR methodology. SCRUM methodology helped with all the management and planning of the artifact.

4. Case Study

The Kimball Methodology and SCRUM methodology were followed for this case study. In this section, all the crucial steps that helped in the development of the OLAP framework are presented.

4.1. Project Planning

For project management, the SCRUM methodology was used to improve project efficiency and reduce the waste of time and resources. First, it was necessary to define and understand the role in the project, then list the priorities, define, and plan the sprints, and later, weekly and daily meetings [26].

For this, the requirements stipulated as mandatory in the patent were also considered; these were the following six layers:

- i. Database(s) serving as the data source, facilitating the provisioning of data to the system;
- ii. An application programming interface (API) responsible for data processing and information provisioning;
- iii. Data warehouse, where the completed multidimensional model is stored;
- iv. OLAP (online analytical processing) layer for data querying and the presentation of diverse perspectives of it;
- v. "Cache" memory layer that manages query storage;
- vi. A visualization layer, which renders the data accessible to end-users through a collection of dashboards integrated within an application.

The sprints in this specific project had a duration of fifteen days. As mentioned earlier (Section 3.4), sprints are based on the product backlog. Consequently, the main tasks to be accomplished were:

- Interpretation of the problem and reality;
- Extracting, loading, and transforming the data;
- Creation of the DW;
- Development of KPIs;
- Development of the OLAP mechanism;
- Extraction of relevant information from the data (data science).

In conclusion, this methodology provides excellent quality, rigor, and efficiency to project management, enhancing quality and agility as a team and developing the solution.

4.2. Project Management

It is expected to demonstrate the efficiency and usefulness of the OLAP system and all activities underlying the process.

4.3. Business Requirements Definition

Functional and non-functional requirements must be met to achieve process excellence. In Table 2, these two types of conditions are displayed. For example, in terms of functional requirements, it is intended to demonstrate a multidimensional analysis, real-time updates, and interactive models and dashboards.

Table 2. Functional and non-functional requirements.

Requirement Type	Requirements
Non-functional requirements	Reliability; Clarity; Efficiency; Ease of Use
Functional requirements	Multidimensional analysis; Real-time updates; Interactive models and dashboards

The activities underlying the realization of these requirements are described in the product backlog in Section 4.1.

4.4. Technology Track

At this stage of the project, it was necessary to carry out the ETL process, KPIs, DW, OLAP, and data analytics. In the ETL stage, pandas library, Microsoft Office Excel, Visual Studio Code, and PAST were used. Likewise, the MySQL workbench was used for developing and loading the DW. The OLAP model was created with cube.js. Cube.js was chosen due to its facility to be included in and interoperable with other solutions, and because it did not require its own platform, as happens for example, with Power BI. More detail is presented in Section 3.2.

4.5. Data Track

The data acquisition process can be conducted using different sensors (magnetic, acoustic/optical, among others). In this case study, the data were provided by the municipal council of Famalicão from underground sensors containing a partition between the entry and exit locations. For parks that do not have a data collection system, it is expected that this project will also develop sensors that collect information from the car parks.

After data collection, a dataset was obtained. That is, the data were analyzed through the excel spreadsheet sent once by the municipal council of Famalicão (with a size of 2347 KB) to better understand the existing attributes. The system is prepared to receive

the data in real time from the API (after finishing the testing phase) and store them in the DW. In this way, in Table 3, it is possible to perceive the attributes obtained, with their description, format, and possible values. For example, the first attribute, Parking lot id, is the number that corresponds to the parking lot, is in integer format, and has values of 1, 2, 3, 4, 5, and 7.

Table 3. Data analyses.

Attributes	Description	Format	Possible Values
Parking lot id	The number corresponding to the lot of the park	Integer	1, 2, 3, 4, 5 and 7
Latitude	Geographic coordinates	Decimal	For privacy reasons, park latitudes will not be shared.
Longitude	Geographic coordinates	Decimal	For privacy reasons, park longitudes will not be shared.
Name	Park Name	String	For privacy reasons, park names will not be shared.
Location	The street where the park is located	String	For privacy reasons, park locations will not be shared.
Displat_id	The number corresponding to the lot from the park	Integer	1, 2, 3, 4, 5 and 7
Provider_id	The number corresponding to the lot from the park	Integer	1, 2, 3, 4, 5 and 7
Parking_cluster_id	-	-	Empty
Parking_cluster_name	-	-	Empty
Parking_cluster_active	-	-	Empty
Provider_type	Sensor from the Dice	Integer	1
Total Capacity	The capacity of places that the park has	Integer	79, 170, 618, 150, 113 and 98
Observations	Zones and sections that the park has	String	"Zona1, Z1 E6-E E5-S Z2 E3-E E4-S" "Zona2 Z2 E3-E E4-S" "Feira Norte: Zona6 Z6 E5-E E6-S Feira Sul: Zona6 Z6 E3-E E4-S" "Zona3 Z3 E3-E E4-S" "Zona4 Z4 E4-E E3-S" Zona15—Z15 E5-E E6-S
Last Update	Time and day when data collection started	Date	Hours and days of February and March 2022
Total Available Places	Total available places that the park had at the time the data was collected	Integer	array([52., 160., 633., ..., 3215., -97., 3220.])
Total Occupied	Total number of occupied places in the park at the time the data was collected	Integer	array([27., 10., -15., ..., -3065., 715., -3070.])
Active	If the park is active	String	Verdadeiro
Update date	Time and date that data collection ended	Date	Hours and days of February and March 2022
Total_places	The capacity of places that the park has	Integer	79, 170, 618, 150, 113 and 98
No_communication_day	If there was a failure in communication in someday	String	False
No_communication_hour	If there was a failure in communication in some time	String	False

Table 4 presents the KPIs identified to quantify the status of the objectives described in Section 4.3. These indicators were defined with the Famalicão municipal council based on the context and analysis of related articles and the previously analyzed dataset (Table 3). An indicator is, for example, the areas with the most movement and the respective KPI is the areas with traffic exceeding two thousand movements per year.

Table 4. Indicators and KPIs.

Number	Indicators	KPIs
1	Areas with the most movement	With traffic exceeding two thousand movements per year
2	Busiest Parks	With traffic exceeding one thousand five hundred movements per month
3	Busiest days	With traffic exceeding one hundred and fifty movements per day
4	Parks with the highest capacity	Capacity greater than three hundred places
5	Stocking rate	Stocking rate greater than fifty percent
6	Hours with the most affluence	With traffic exceeding two hundred movements per hour

4.5.1. ETL Process

The ETL process is critical to processing the data and improving the performance of the solution to be developed. The dataset was analyzed at this stage through the pandas library. Functions were created and methods were used to find null and unique values, separate the times and dates into two columns, and understand the sample's representativeness. After this analysis, writing errors were corrected through the creation of a dictionary for park names; null values were removed from the data sample and the creation of two different columns for times and dates so that in the data warehouse, there were two distinct dimensions. For example, one of the locations was spelt in two different ways, "Rua Fernando Mesquita" and "Rua Farnando Mesquita", but both streets refer to the same location.

A new column, "Dif", was added through the Python language, representing the difference between the attributes total_available_places and total_occupied_places. Then a function was created that, whenever the difference values change, registers 1 in a new column called "Mov". If the different values change, there is movement in the park. Otherwise, the value of 0 is recorded in the new column.

After a detailed analysis of the dataset, a selection of columns was made considering the indicators and defined KPIs (Table 4). That is, columns that are not significant and not relevant for future analysis were excluded: displat_id, provider_type, provider_id, active, parking_cluster_id, parking_cluster_name, parking_cluster_active, total_places, no_communication_day, and no_communication_hour.

4.5.2. Data Warehouse Development

According to the data (Table 3), it was possible to perceive that it was necessary to build two fact tables, one relative to the movements and another to the data of each park. Considering that the dataset provides hours and days of each movement, two dimensions, calendar and time, were built. The latitude and longitude attributes allowed the creation of a dimension called location. Furthermore, the attribute with the name of the park's street allowed the creation of the local dimension.

After identifying the fact tables and dimensions, the model was built, and the possible connections between the elements were identified. In this way, Figure 4 shows that the availability management fact table, which refers to the park's movements, is linked to the

calendar and time dimensions. Moreover, the fact table relating to park management is connected to the location and local dimension.

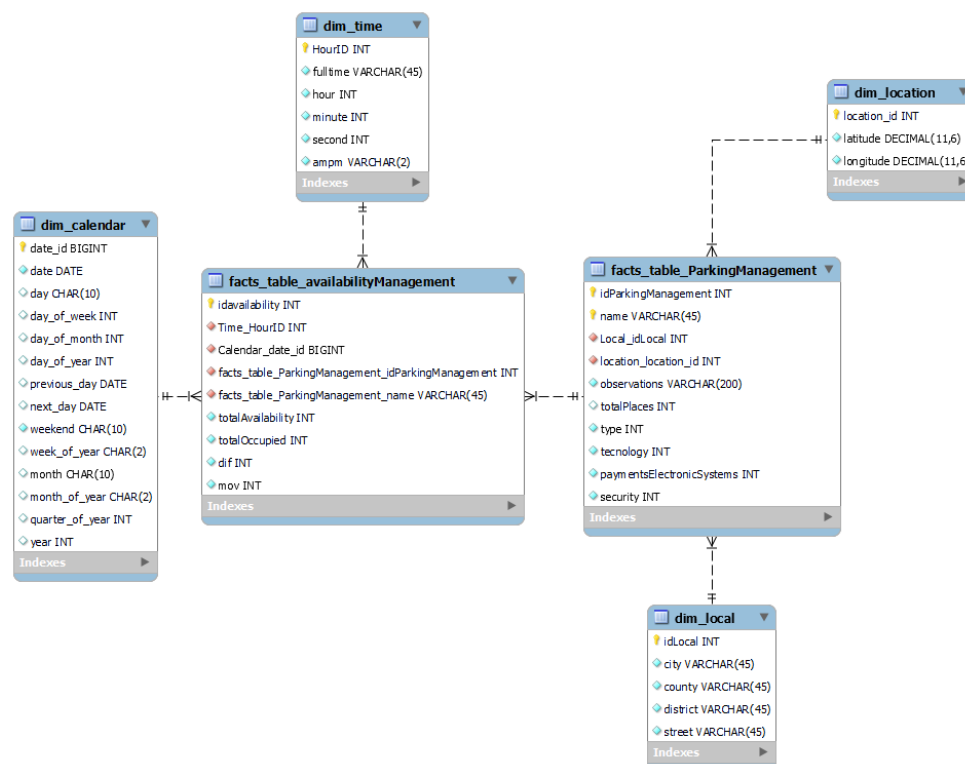


Figure 4. Data warehouse.

Park identifier data, such as park name and street, were anonymized to ensure information privacy.

In this phase, the data warehouse was created with four dimensions (time, calendar, local, and location) and two fact tables (availability management and parking management). These dimensions and fact frameworks were designed and structured based on the analysis of the obtained data. In other words, as we had information on time in the dataset, it was possible to create two dimensions, one related to hours and another related to days. In the time dimension, the attributes are id, full-time, hour, minute, second, and whether it is am or pm. In the calendar dimension, features include the id, the complete date, the day of the week, the month, and the year, whether it is a weekend or day of the week, month and month number, and year, as well as the date of the previous day and the date of the next day.

The dataset's information on the street name made it possible to determine the park's county, district, and city, creating the local dimension.

The location dimension was created with the information of geographic coordinates (latitude and longitude).

The parking management fact table refers to the unique data and information on the existing car parks. As all parks have geographic coordinates and an associated location, there is a one-to-many connection between the location and local dimensions, so the fact table inherits this data from the park. It also has attributes related to the lot, name, observations, and total park places.

The availability management fact table refers to the movements that each park has. The table is linked to the time and calendar dimension since each movement is associated with a time and a date. It is also linked to the parking management fact table to identify and inherit the parking associated with the movement.

The columns of interest were filtered, and after the necessary changes, all selected data were imported for each dimension of the data warehouse. All dimensions were populated through MySQL Workbench. The dimensions were filled using MySQL scripts. The fact tables were populated using the pandas library (open-source software for data analysis and manipulation). Fact tables were populated using this library as it was necessary to extract data from Excel and aggregate information from the associated dimensions. This was the most viable approach to incorporate all the entered data to ensure their alignment. The data warehouse was populated once with the processed data.

4.6. Business Intelligence Track

After the data warehouse was developed and loaded with data, selecting the application that supported the project's requirements was necessary. Cube.js was the chosen data visualization tool because it has various advantages favoring project development.

Cube.js is an open-source headless BI that allows pre-calculation and pre-aggregation of data, creating an OLAP cube that makes analysis faster. The cube.js enables the connection to several databases, and consequently, the schema is defined for each dimension and fact table in several cubes. For the fact tables, the foreign keys and relationships with other cubes are defined, as well as the facts that will be calculated. Through pre-aggregations, we can save the desired combinations for faster analysis. The goal is to obtain analytical dashboards based on the implemented DW and materialized views that satisfy the business management's information needs [2].

4.7. Deployment

Initially, the data warehouse was linked to cube.js. Next, the data schema was developed for each dimension and fact table. The primary key and the respective columns were specified in the dimension's schema. Furthermore, the primary keys, respective columns, foreign keys, and facts (measures) were identified for the fact tables schema. After the model's data schema was specified, pre-aggregations were made to respond to the KPIs identified in Section 4.5. Finally, the statistical data can be viewed in the playground section of cube.js. Figure 5 displays the workflow in cube.js. Cube.js is a software solution that offers an API that will allow the use of this mechanism in any other system or interface. In simpler terms, cube.js offers a convenient way to access and utilize its capabilities through an API. This API communicates with the cube.js backend, and all requests made to the backend will be presented in the developed frontend. In addition, the OLAP mechanism has an API behind it that allows adaptation to new realities and the creation of new indicators.

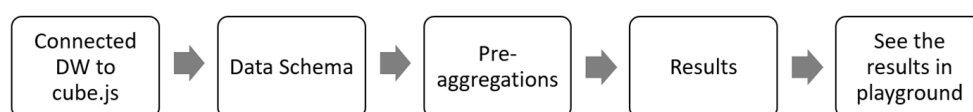


Figure 5. Workflow in cube.js.

In conclusion, the traditional BI solution did not meet all the requirements, leading to the development of a new OLAP framework.

5. OLAP Framework

The DSR methodology helped to understand the problem and define the solution. Thus, the developed artifact (OLAP framework) is presented in this section.

5.1. Identifying the Problem and Motivation

This project arises from the need to change mobility alternatives due to excess congestion, pollution, and accidents. There is a need to demonstrate the data in an intuitive and easy-to-use application to support decision making. A schema was developed in cube.js to create an OLAP model by analyzing a dataset and data warehouse. However, despite

cube.js being quite intuitive, there are fundamental characteristics in a three-dimensional analysis that it does not present, namely:

- The drill-down and roll-up processes;
- Various types of graphs;
- Data suggestion in the filtering process;
- Possibility of saving the charts created;
- Intuitive and short labels;
- A modular, scalable, and adaptable solution to any industry.

In short, the motivation arose to develop an OLAP framework that can respond to the requirements.

5.2. Definition of Solution Objectives

The main objective is to develop an artifact that responds to the requirements mentioned in step 1 of the DSR methodology.

5.3. Design and Implementation

In this phase, the intended architecture for the OLAP framework was built. After researching the most appropriate technologies for developing this layer, the following languages and tools were applied to create the solution.

Figure 6 presents part of the general architecture of the ioCity project. This architecture shows the development flow to the OLAP framework's desired solution, which contains three layers represented in the image: knowledge, API, and interface.

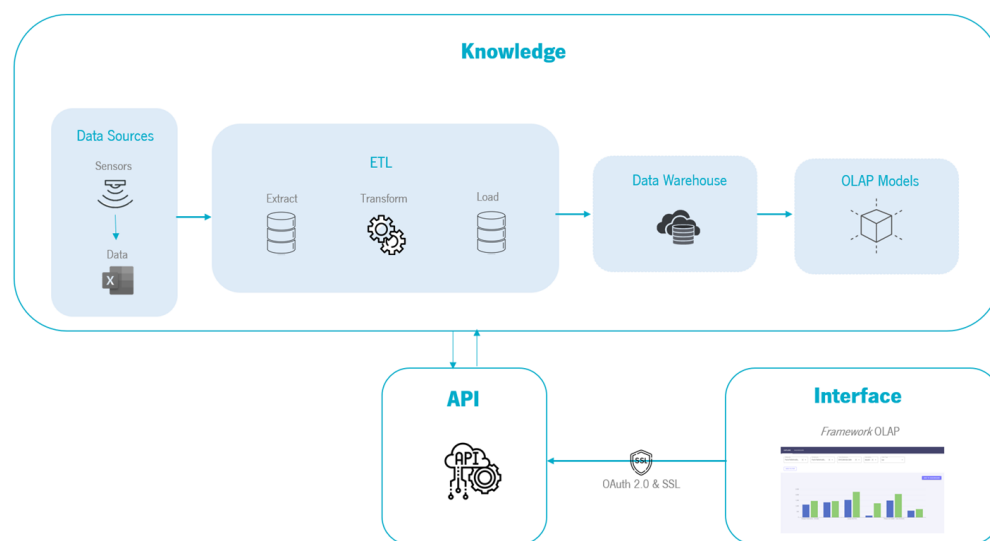


Figure 6. OLAP framework architecture.

Regarding the knowledge layer:

- From the data obtained (data sources), the ETL process was started—extraction, transformation, and loading, where data cleaning was performed. Consequently, the data warehouse was designed to store all the data. The OLAP model was created with the cube.js tool.

Interface layer:

- With the cube.js API, an OLAP framework was developed to visualize and create interactive graphs that help decision making. The solution is scalable, and the data indicate what is possible.

The API layer interconnects the two layers (knowledge and interface).

Vue.js was used for frontend development. Vue.js is a JavaScript framework for developing reactive web interfaces that extends HTML and CSS languages for frontend

construction. Through the cube.js API, the entire OLAP model was inherited. A graphical user interface library called Vuetify was used to develop a more appealing and intuitive frontend. Apache ECharts was needed when creating the data visualization graphs for the user, and the GoogleMaps API was used for making the map graph.

A backend API has also been incremented to define data routes via MySQL. For this, an API was created with node.js/express.js (backend framework for developing APIs that facilitates the creation of web pages with node.js) to obtain the necessary data. node.js is open-source software that allows one to create JavaScript applications without relying on a browser. All the code was designed and developed in the Visual Studio Code tool to achieve the project's objectives. For more detail, see Section 3.3.

5.4. Demonstration

The created OLAP framework has three main pages. The entire solution meets the defined objectives, and the tools and languages are described in the previous subsection. In this way, the answer will be presented according to the main pages: Explore, Dashboard, and KPIs, where all the features will be described.

- Explore

On this page (Figure 7), it is possible to explore different data combinations between facts, dimensions, and temporal dimensions with granularity through various graphs. The graphs that allow granularity are line, area, bar, and radar graphs, while pie, scatter, doughnut, and map graphs do not allow this type of analysis. It is also possible to add filters and save the chart on the Dashboard page associated with a name. The filtering process encompasses all dimensions and facts in the data warehouse and various operators for exploring the data. On this page, it is also possible to view an auxiliary table for the graphs with the respective data of the created chart, both in percentage and value terms.

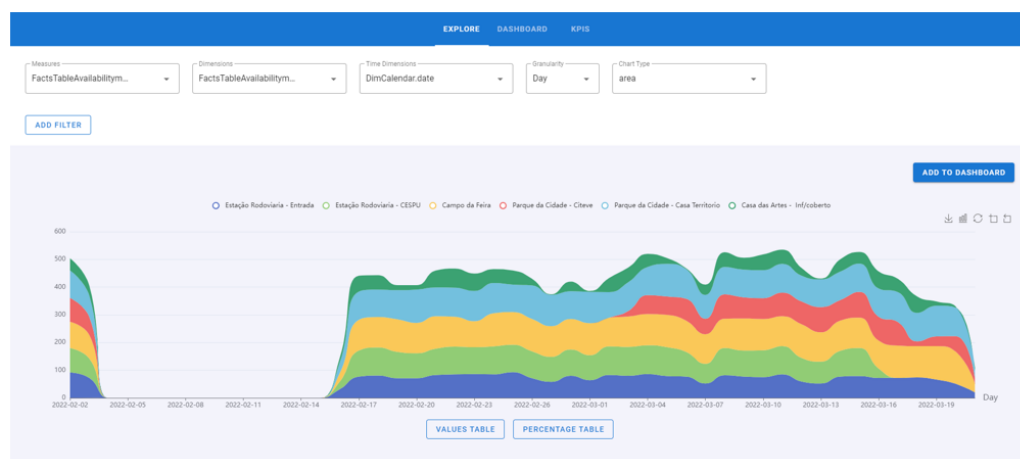


Figure 7. Explore page.

In this figure, it is possible to see a graphic with three dimensions, where the x-axis represents the days of the year 2022 and the y-axis represents the values of movements. In this way, this graphic demonstrates the total movements per day for each park. The data presented are simulated to preserve their anonymity.

- Dashboard

All saved graphs are displayed on this page (Figure 8). Graphics that allow for granularity feature drill-down and roll-up by time and location. By default, when clicking on a bar/line/area, the drill-down process will be displayed, and after the end of the process, the possibility of returning to the initial chart or roll-up is presented. The map chart also allows the drill-down and roll-up processes, but zooming enables the visualization of more detailed data. In contrast, the less the degree of zooming, the fewer facts are presented

as the information is grouped. The filtering process is available on this page and applies to all graphs. The same is used if the filter is applied to any saved graphs. Otherwise, the chart remains the same. The possibility of deleting a graphic is also available through the “Delete” button. At this moment, two graphs exist on the Dashboard page: on the left side is a graphic representing each park’s movements in March; on the other side is a graphic representing each park’s weekly movements.

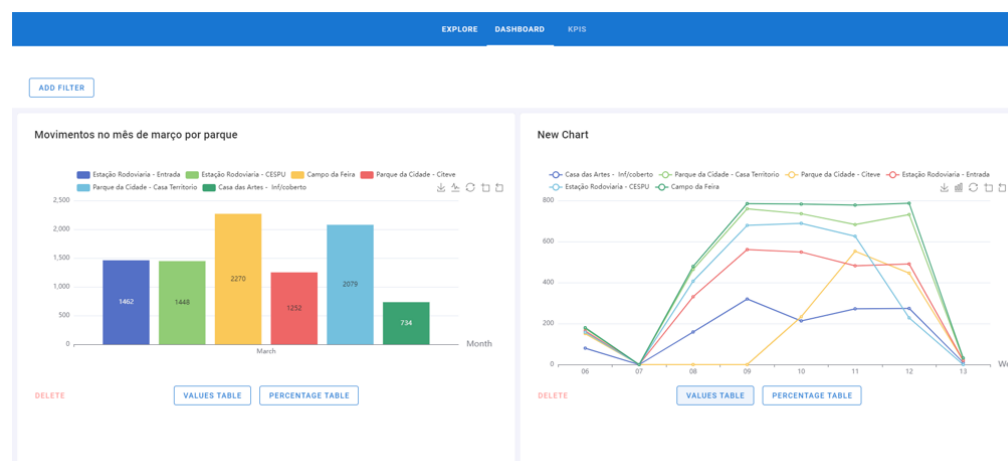


Figure 8. Dashboard page.

- KPIs

KPIs are displayed in several charts (Figure 9). A set of KPIs was defined to quantify the status of the objectives. These indicators were determined based on the context and analysis of related articles. An indicator is, for example, the areas with the most traffic and the respective KPI is the areas with more than 2000 traffic movements per year. The KPIs were described in detail in the article “Intelligent dashboards for car parking flow monitoring”.



Figure 9. KPI page.

5.5. Evaluation

Several team tests were conducted to evaluate the developed solution, understand its versatility and usefulness, and verify that all requirements were successfully met.

The architecture described in Section 3.1.2 was followed for the artifact development, and the methodology is presented in Section 3.1.

A practical analysis example will be presented to complement the description of the subsection of activity 4 of the DSR (demonstration). This example will demonstrate how making a graph with a three-dimensional analysis associated with a filtering process and temporal drill-down is possible.

The presented data were exposed and analyzed in Section 4.5.

1. Select a fact, dimension, time dimension and chart type (Figure 10).

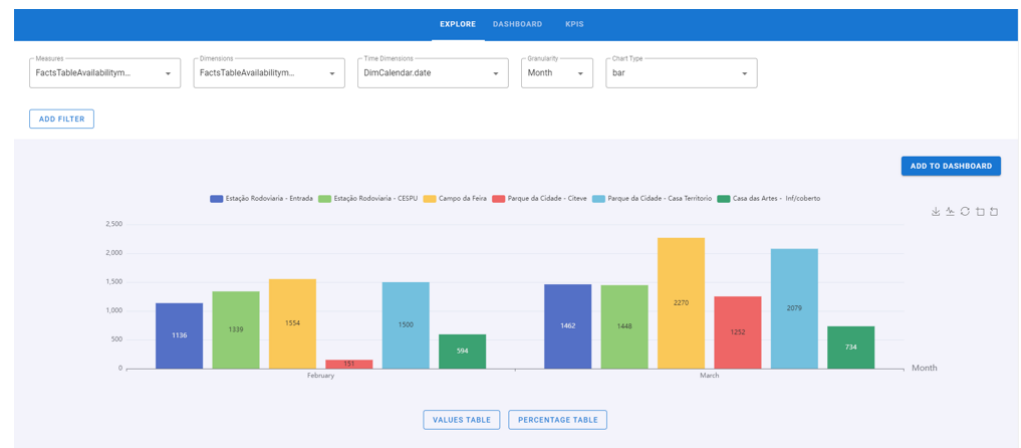


Figure 10. Step 1 of using the OLAP framework.

In this example, select movements for the fact (measures), the park's name for the dimensions, a granularity by month, and bar chart type.

The x -axis represents the months (February and March), and the y -axis is the value associated with each park about movements. For example, the park “Estação Rodoviária—entrada” had 1136 movements in February.

2. After creating the graph, it is possible to visualize a supplementary table to the chart, both in value (Figure 11) and percentage (Figure 12) terms.

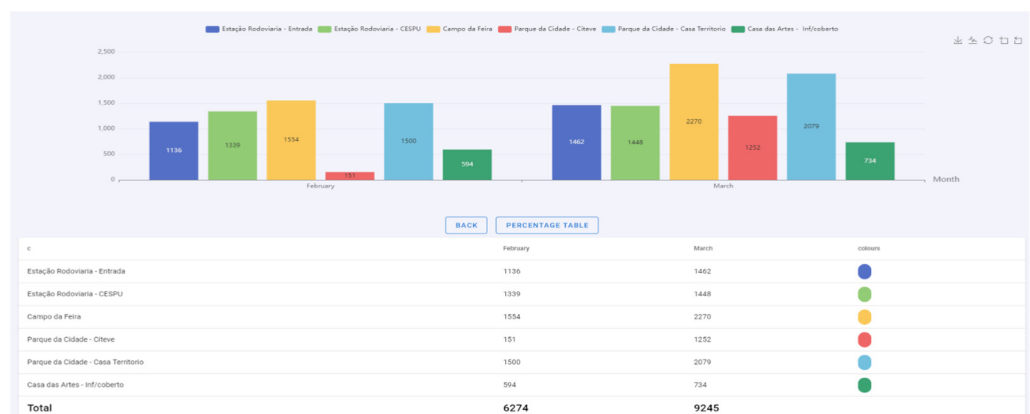


Figure 11. Step 2 of using the OLAP framework.

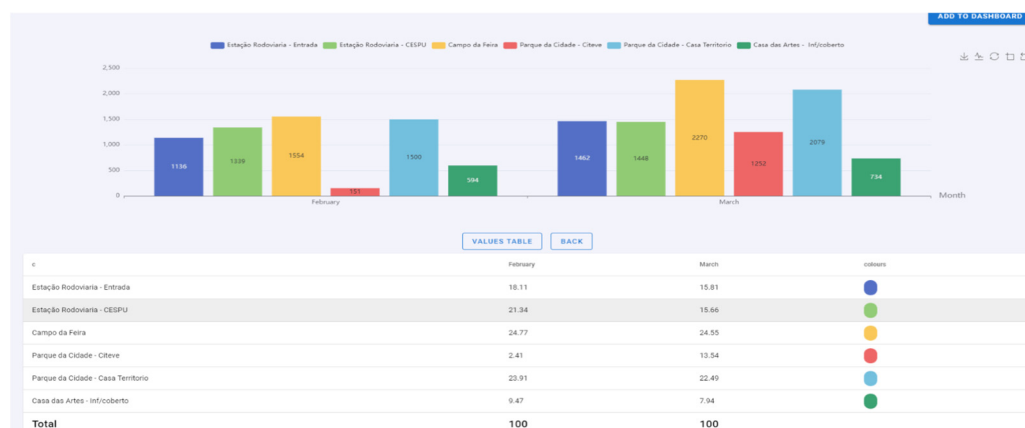


Figure 12. Step 3 of using the OLAP framework.

For example, the park “Estação Rodoviária—entrada” had 1136 movements in February and 1462 in March. This park is represented by blue color in the graphic. The last line is about the total movements for each month.

In the following figure, the auxiliary table presents the values in percentage (Figure 12).

For example, the park “Estação Rodoviária—entrada” only had 18.11% of the movements in comparison to the other parks in February.

- Click on the “Add filter” button. Select a dimension or fact. Select an operator (equals, does not, >, <, =, >=, <=). Insert a value if a fact has been selected or select an attribute if a dimension has been established (Figure 13).

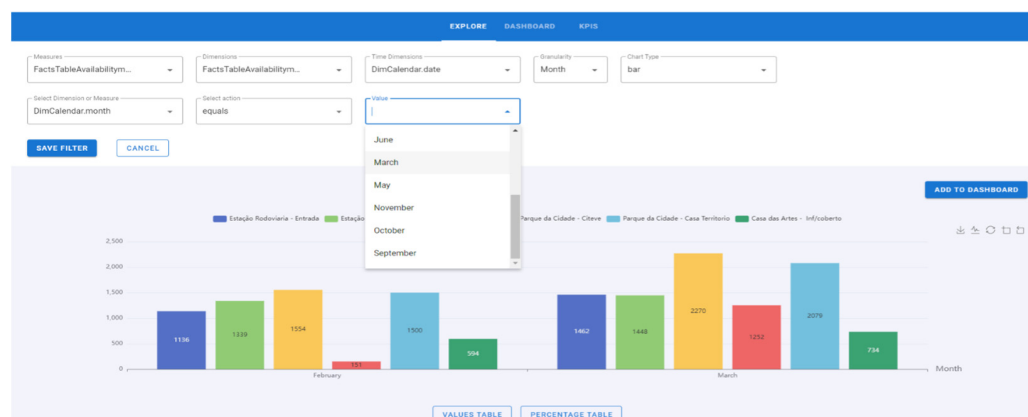


Figure 13. Step 4 of using the OLAP framework.

The attribute month, operator “equal”, and March as value were chosen for the filter. Therefore, when applying this filter, only the movements for March should appear.

- Click on the “Save Filter” button. Several filters can be inserted and later deleted (Figure 14).

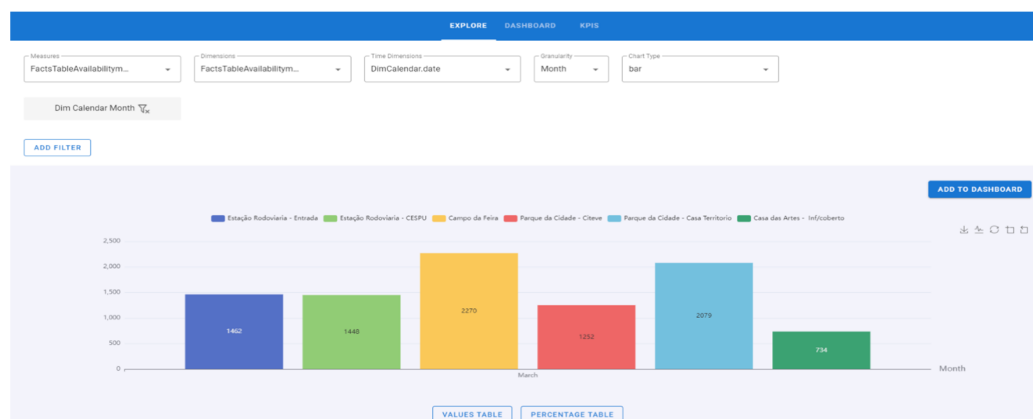


Figure 14. Step 5 of using the OLAP framework.

- Click on the “Add to Dashboard” button, give the chart a name and click on “Save” (Figure 15). The name of the graph is “Movements in March by the park” (represented in the figure by “Movimentos no mês de março por parque”).

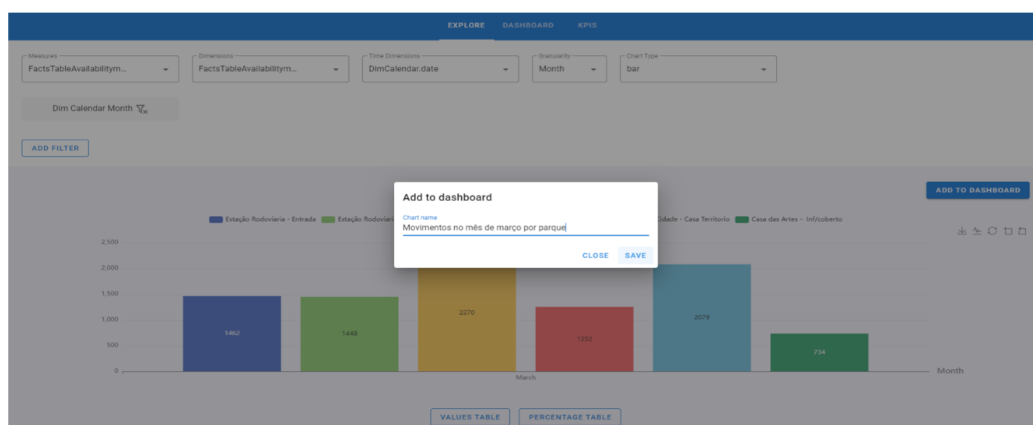


Figure 15. Step 6 of using the OLAP framework.

- The chart is already on the Dashboard page and can be deleted (Figure 16). The graphic added at this time is on the left side. On the right side appears a graphic that had previously been added.

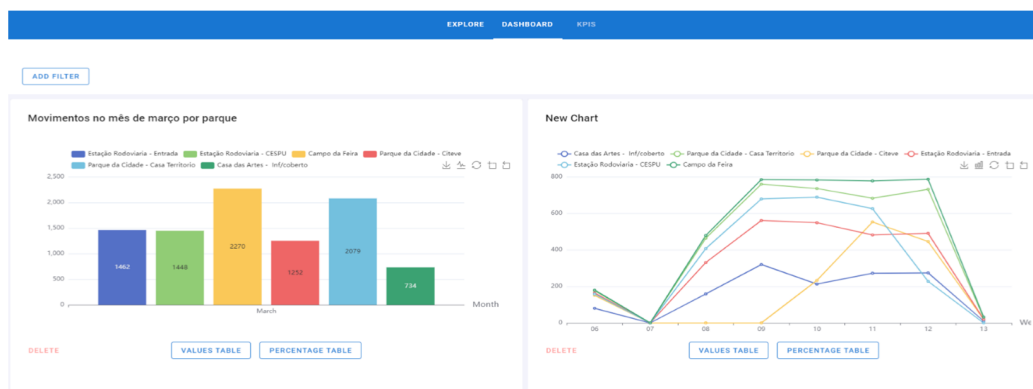


Figure 16. Step 7 of using the OLAP framework.

7. To perform the drill-down process, step 1 must be strictly followed. In this process, click on the bar/line/area and the graph will go down a level. During the drill-down, the degree where the process is located is displayed, that is, if the chart is per year, quarter, month, weeks, days of the week, or hours, and the attribute previously selected. In this example, the “Campo da Feira” bar with 2270 movements was set (Figure 16).

Consequently, the chosen degree and the respective attribute (previous degree) appear: month of March and the time period currently shown on the graph: week. One new chart per week is displayed, with only weeks 10, 11, 12, and 13 showing movement (Figure 17).

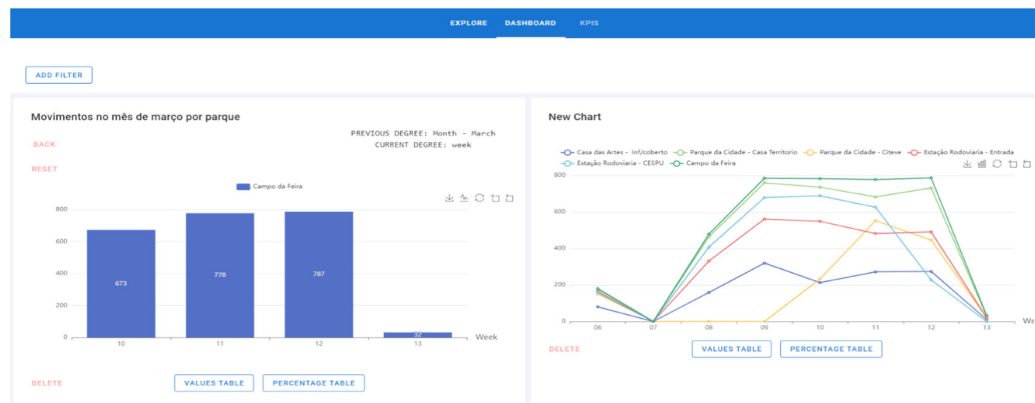


Figure 17. Step 8 of using the OLAP framework.

- a. Week 12 in Figure 17 has been selected. A chart by days is displayed. In this case, there are only movements on 20 March 2022 and 21 March 2022. The bar corresponding to 20 March 2022 was selected (Figure 18).

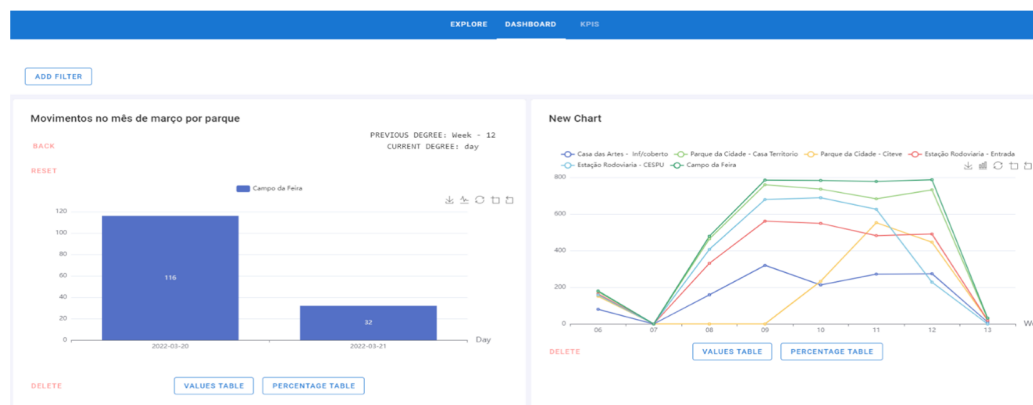


Figure 18. Step 9 of using the OLAP framework.

- b. The hours when there was movement on that day (20 March 2022) are displayed. As hours are the lowest attribute of the drill-down, the process ends (Figure 19).

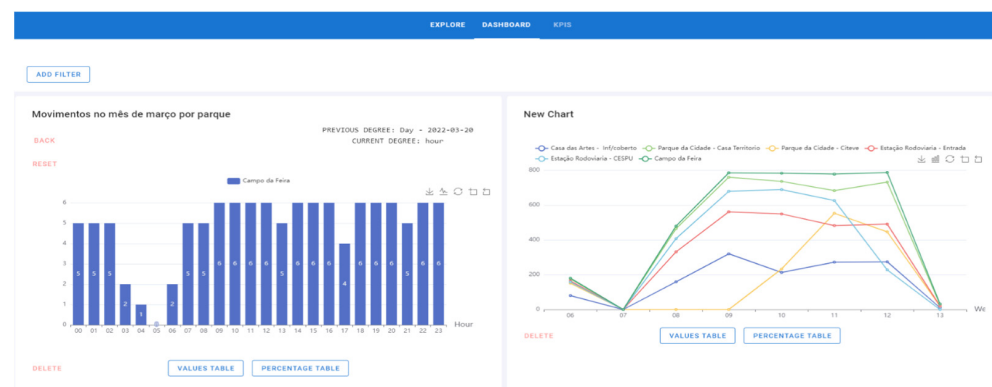


Figure 19. Step 10 of using the OLAP framework.

8. When the drill-down is finished, an end message is displayed. As soon as the drill-down is performed, a “Reset” button is displayed to return to the initial chart. There is also the possibility of doing a roll-up (opposite of the drill-down process) through the “Back” button. That is, instead of the degree going down (year–quarter–month–weeks–days–hours), it will go up; so in this temporal example, the flow will be hours–days–weeks–month–quarter–year. The year is the highest level of the process (Figure 20).



Figure 20. Step 11 of using the OLAP framework.

9. In this case, if the user wants to go back to the previous graph, it is only necessary to click on the “Back” button and the chart will update to the last degree; in this example, for the days of the week (Figure 21). That is, after pressing the button, the graph is displayed and the labels are updated, showing the previous degree: hours (last degree: hour) and the current degree: days (current degree: day).

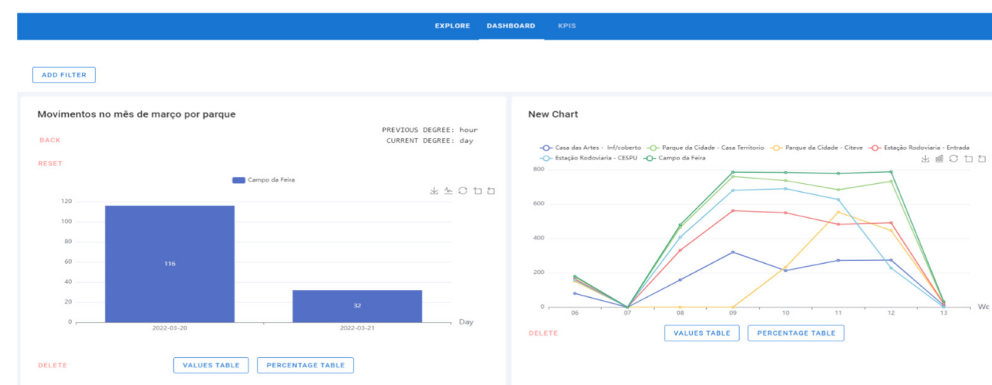


Figure 21. Step 13 of using the OLAP framework.

10. Finally, if desired, the original graphic will be displayed by pressing the “Reset” button (Figure 22). The filtering process can also be included if desired.

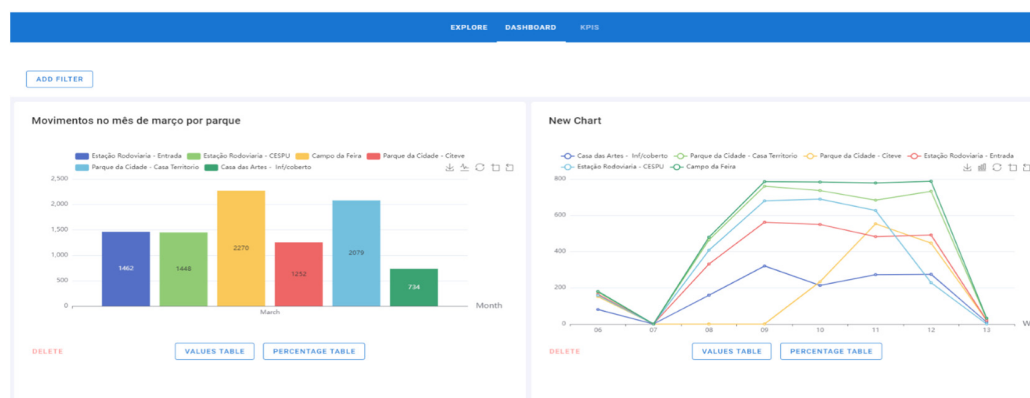


Figure 22. Step 14 of using the OLAP framework.

11. The page with all the KPIs is presented (Figure 23). The KPIs represented here can be found in Table 4.

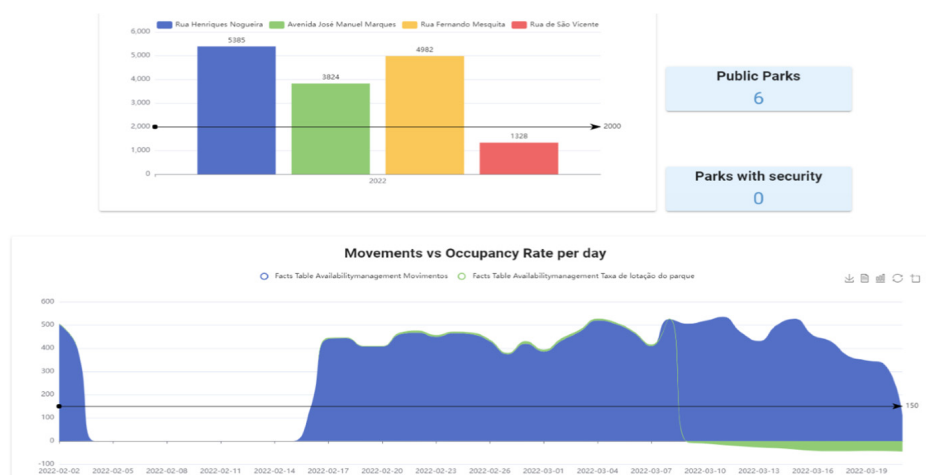


Figure 23. Step 15 of using the OLAP framework.

5.6. Communication

In this phase, three articles were written and published. However, regarding the OLAP model, the article “Intelligent dashboards for car parking flow monitoring” aimed to demonstrate the importance of an OLAP model in decision making and how the entire flow until the construction of the model is fundamental for a valuable and effective model (Section 2.8). This article is also part of the communication.

6. Discussion

The entire development process, from data analysis to the design of the OLAP framework, is crucial to obtain good results.

Through the OLAP process, it is possible to understand several metrics and indicators useful in decision making. This analysis of the current context allows better park management, creating more value and usefulness for those who enjoy and manage the parking. Metrics such as crowding rate, movement, and busiest hours can be analyzed. For example, suppose there is high demand compared to the available spaces on a particular street. In that case, constructing a new car park can be an option to create more parking spaces and reduce traffic flow, allowing for better mobility.

The need to build a new OLAP framework arose because cube.js has some flaws that the team considers relevant in an OLAP analysis. In this sense, Table 5 maps the functionalities between cube.js and a developed OLAP framework, where X means (has/includes).

Table 5. Cube.js vs. OLAP framework.

Features	Cube.js	OLAP Framework
Define data structure	X	
Data pre-aggregation	X	
Possibility of connection to multiple data warehouses	X	
Possibility of several dice combinations	X	X
Filtering process	X	X
Suggestion of data in filtering process		X
Various graphics of visualization and map		X
Auxiliary table for the graphics with numeric values or percentages		X
Drill down		X
Roll-up		X
Page to save graphics useful for taking decision		X
Short chart labels and objective		X
Save graphics for use particle		X
KPI views		X

The OLAP framework is highly scalable and easily adaptable to other datasets, allowing a three-dimensional analysis that helps decision making. Ten crucial features to support decision making were identified, and a data warehouse, an OLAP model, and six indicators were built for the developed solution. This OLAP framework has three components: creating charts, and analyzing and visualizing the data. In Table 6, the functionalities of the solution are shown. The first feature is the possibility of combining data across three dimensions. The significant advantage compared to other existing tools, such as Tableau and Power BI, is that this framework is adaptable to any industry. Its integration into software as a service (SaaS) is available for analysis without the internet and can be fully tailored to the user.

Regarding the patent, this framework is modular and allows for the integration of different types of data sources, including structured and non-structured data, as well as offline data analysis. For example, the case study has two distinct databases, PostgreSQL and Mongo DB, and the KPIs can be consulted without any online connection, with the data being updated when the connection is restored. As it is modular, the data visualization module of the framework will be incorporated into the ioCity platform, and the other modules are centrally executed by ioScience [3,4].

Table 6. OLAP framework functionalities.

Number	Functionality	Description
1	Combine data	A three-dimensional analysis from multiple perspectives is possible.
2	Add filter	After creating the graph, the user can apply filters on the chart and analyze and explore the subset of data presented, drawing conclusions that can help the decision-making process.
3	Suggest data in the filtering process	During the filtering process, choosing a dimension or fact for the filter and a suggestion of operators to use is possible—for example, greater, lesser, or equal. Finally, if the filter has one dimension, a dataset will be suggested to the user. If the filter is for a fact, the user can enter the value he wants (since the facts are measurable, the user can choose the value he wants).
4	Create many types of charts and a map chart	The developed solution allows the user to create bar, line, area, radar, scatter, doughnut, and circular graphs. Creating a map chart is also possible.
5	View data in two formats (numerical and percentage)	All charts have an auxiliary table for a better reading of the data. The table can be viewed with absolute or percentage values.
6	Drill-down	The user can, except for pie charts and doughnuts, drill down by clicking on a point on the chart. This process helps the user to view the data in greater detail at various hierarchical levels. The map chart allows the drill-down process via zoom, so the more significant the zoom at a given point, the greater the detail.
7	Roll-up	The roll-up process is the opposite of the drill-down process and helps the user analyze the data more globally.
8	Save charts for decision making	All graphs created can be saved on the Dashboard page for further analysis.
9	Short and objective labels	Through the graphics created, the user can quickly understand the context of each one due to intuitive and quick-to-read captions.
10	Download charts	Possibility of downloading graphics for the user's private use.

7. Conclusions

All of the developed work involves several components, such as data science, web development, and data visualization. The first component involves all data processing and analysis to extract meaningful information and identify decision-making patterns. To develop the solution, it was necessary to create the web (backend API to define data routes via MySQL) and the entire data visualization component to clearly and advantageously make all the information available to the user.

The developed OLAP framework provides the user with a complete analysis of the data, supporting them in the decision-making process. All the steps until the construction of the OLAP model are essential for the final result to be of quality and value. The process becomes efficient with well-structured data and an understanding of its context. Thus, an analysis of the data was carried out, and consequently, a data warehouse was created to identify measurable KPIs (six KPIs were identified).

This work presents an essential contribution to IoT projects, although the framework is characterized by its versatility and adaptability to any industry. Through the models and the developed solution, it is possible to analyze several factors that citizens prioritize and, consequently, analyze the data (from the sensors) to draw meaningful conclusions for decision making. It is possible to objectively and helpfully check in detail all the data collected, thereby ensuring that the digital transformation in cities adds significantly more value and meets the needs of the habitants.

The paper makes two contributions: the global framework that any industry can use, and the OLAP model and respective dashboards that can be used to monitor city parking movements. Both contributions can be easily replicated following Sections 4 and 5 and explored using any dataset.

The framework successfully met all identified requirements and fully supported the patent guidelines, achieving a 100% effectiveness and efficiency rate in this case study. This demonstrates the cache's relevance, as all tasks were carried out successfully and instantly.

The project is divided into several phases; in this phase, all objectives were successfully achieved. Concerning future stages, improvements in terms of data visualization will be made so that the final product is ready for use and more intuitive. Future work will aim to optimize the framework and include predictive models.

Future Work

In a future phase of this study, the possibility of automating the modeling process using agents and artificial intelligence (AI) and agent-based modeling, which can better analyze the problem and provide a future process, can be considered as potential work. Moreover, we could also analyze future similar frameworks, increase the size of the dataset used for testing efficiency and greater speed, and implement it in a wider range of industries.

Author Contributions: Conceptualization, F.P. and F.B.; methodology, F.P.; software, F.B. and B.R.; validation, J.V. and F.P.; formal analysis, F.P.; investigation, F.B.; resources, F.P. and J.V.; data curation, F.B. and J.V.; writing—original draft preparation, F.B.; writing—review and editing, B.R. and F.P.; visualization, F.B. and B.R.; supervision, F.P.; project administration, F.P.; funding acquisition, F.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work has also been developed under the scope of the project NORTE-01-0247-FEDER-045397, supported by the Northern Portugal Regional Operational Programme (NORTE 2020), under the Portugal 2020 Partnership Agreement, through the European Regional Development Fund (FEDER).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Onesmus Mbaabu. MOLAP vs ROLAP vs HOLAP in Online Analytical Processing (OLAP). Engineering Education (EngEd) Program | Section 2021. Available online: <https://www.section.io/engineering-education/molap-vs-rolap-vs-holap/> (accessed on 7 June 2022).
2. Cube Cube—Headless BI for Building Data Applications. CubeDev 2022. Available online: <https://cube.dev/> (accessed on 7 June 2022).
3. Portela, F.; Fernandes, G.; Alves, P.; Ferreira, J.A. *Method to Execute Offline Data Analysis*; Portugal PT. ID. 116393, IPC: G06F 16/00 (2019.01); IOTCHPIS-Innovation on Technology, Lda.: Braga, Portugal, 2022.
4. Fernandes, G.; Portela, F.; Santos, M.F. Towards the Development of a Data Science Modular Solution. In Proceedings of the 2019 7th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), Istanbul, Turkey, 26–28 August 2019. [CrossRef]
5. ioCity. Available online: <https://iocity.research.iotech.pt/> (accessed on 28 December 2022).
6. Francisca, B.; Beatriz, P.; Vasco, P.; José VCarlos, F.; Filipe, P. *An analysis of Smart Cities Initiatives in Portugal*. Springer EAI—Sustainable, Innovative and Intelligent Societies and Cities; Springer: Berlin/Heidelberg, Germany, 2022.
7. Mesároš, P.; Behúnová, A.; Mandičák, T.; Behún, M.; Krajníková, K. Impact of enterprise information systems on selected key performance indicators in construction project management: An empirical study. *Wirel. Netw.* **2021**, *27*, 1641–1648. [CrossRef]
8. Batty, M.; Axhausen, K.W.; Giannotti, F.; Pozdnoukhov, A.; Bazzani, A.; Wachowicz, M.; Ouzounis, G.; Portugali, Y. Smart cities of the future. *Eur. Phys. J. Spec. Top.* **2012**, *214*, 481–518. [CrossRef]
9. Wang, Y.; Kung, L.; Byrd, T.A. Big data analytics: Understanding its capabilities and potential benefits for healthcare organizations. *Technol. Forecast. Soc. Change* **2018**, *126*, 3–13. [CrossRef]
10. TOTVS Análise de dados: O que é, Vantagens, Tipos e como Fazer. 2022. Available online: <https://www.totvs.com/blog/negocios/analise-de-dados/> (accessed on 14 June 2022).
11. Karagiannis, A.; Vassiliadis, P.; Simitsis, A. Scheduling strategies for efficient ETL execution. *Inf. Syst.* **2013**, *38*, 927–945. [CrossRef]
12. IBM Cloud Education What is ETL (Extract, Transform, Load)? 2022. Available online: <https://www.ibm.com/cloud/learn/etl> (accessed on 14 June 2022).

13. Hwang, M.I.; Xu, H. A Structural Model of Data Warehousing Success. *J. Comput. Inf. Syst.* **2008**, *49*, 48–56. [CrossRef]
14. Anggrainy, T.D.; Sari, A.R. Implementation of Extract, Transform, Load on Data Warehouse and Business Intelligence Using Pentaho and Tableau to Analyse Sales Performance of Olist Store. *Science* **2022**, *7*, 368–374.
15. What Is Power BI? Definition and Features | Microsoft Power BI. Available online: <https://powerbi.microsoft.com/en-us/what-is-power-bi/> (accessed on 23 December 2022).
16. MultiTech. What Are the Advantages and Dis Advantages in Power BI. Medium 2020. Available online: <https://informationit27.medium.com/what-are-the-advantages-and-dis-advantages-in-power-bi-98b49152daf4> (accessed on 23 December 2022).
17. Lu, M. Discovering Microsoft Self-service BI solution: Power BI. 2014. Available online: <https://www.theseus.fi/handle/10024/76619> (accessed on 23 December 2022).
18. What Is Tableau? Available online: <https://www.tableau.com/why-tableau/what-is-tableau> (accessed on 23 December 2022).
19. Advantages and Disadvantages of Tableau. Available online: <https://absentdata.com/advantages-and-disadvantages-of-tableau/> (accessed on 23 December 2022).
20. Leite, N.; Pedrosa, I.; Bernardino, J. Open Source Business Intelligence on a SME: A Case Study Using Pentaho. In Proceedings of the 2019 14th Iberian Conference on Information Systems and Technologies (CISTI), Coimbra, Portugal, 19–22 June 2019; pp. 1–7.
21. de Sá Mussa, M.; de Souza, S.C.; da Silva Freire, E.F.; Cordeiro, R.G.; da Hora, H.R.M. Business intelligence na educação: Uma aplicação do software pentaho. *Rev. Produção Desenv.* **2018**, *4*, 29–41. [CrossRef]
22. Barros, F.; Fernandes, C.; Portela, F. Opinion Clustering About Mobility Decisions. In *Advanced Research in Technologies, Information, Innovation and Sustainability. ARTIIS 2022. Communications in Computer and Information Science*; Guarda, T., Portela, F., Augusto, M.F., Eds.; Springer: Cham, Switzerland, 2022; Volume 1675. [CrossRef]
23. Hammad, A.; AbouRizk, S.; Mohamed, Y. Application of KDD Techniques to Extract Useful Knowledge from Labor Resources Data in Industrial Construction Projects. *J. Manag. Eng.* **2014**, *30*, 05014011. [CrossRef]
24. Peffers, K.; Tuunanen, T.; Rothenberger, M.A.; Chatterjee, S. A Design Science Research Methodology for Information Systems Research. *J. Manag. Inf. Syst.* **2007**, *24*, 45–77. [CrossRef]
25. Kimball DW/BI Lifecycle Methodology. Available online: <https://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dw-bi-lifecycle-method/> (accessed on 14 June 2022).
26. Builder, P. Entenda a metodologia Scrum e como utilizá-la na empresa. Project Builder 2020. Available online: <https://www.projectbuilder.com.br/blog/o-que-e-scrum/> (accessed on 27 April 2022).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.