Fleet and traffic management systems
for conducting future cooperative mobility

D3.1 Specification of data gathering harmonization, fusion and analysis techniques

Document Type	Deliverable
Document Number	D3.1
Primary Author(s)	Miha Cimperman, Luka Stopar, Matej Polzelnik | Jožef Stefan Institute
Document Version / Status	V1.0| Final
Distribution Level	PU (public)

Project Acronym	CONDUCTOR
Project Title	Fleet and traffic management system for conducting cooperative mobility
Project Website	https://conductor-project.eu/
Project Coordinator	Netcompany Intrasoft SA | www.netcompany-intrasoft.com
Grant Agreement Number	101077049

CONDUCTOR project has received funding from the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101077049.

CONTRIBUTORS

Name	Organization	Name	Organization
Miha Cimperman	JSI	Raquel Sánchez	Nommon
Luka Stopar	JSI	Pablo Ruiz	Nommon
Matej Polzelnik	JSI	Oliva García Cantú	Nommon
Arslan Ali Syed	TUM	Panagotis Georgakis	Frontier
Pantelis Lappas	INTRA	Konstantinos Katzilieris	NTUA
Konstantinos Chasapas	INTRA		
Nasos Grigoropoulos	INTRA		

FORMAL REVIEWERS

Name	Organization	Date
Nasos Grigoropoulos	INTRA	2024-01-15
Oskar Eikenbroek	UTwente	2025-01-17

DOCUMENT HISTORY

Revision	Date	Author / Organization	Description
0.4	2023-10-25	Miha Cimperman (JSI), Luka Stopar (JSI)	Definition of terms of content
0.5	2023-12-01	Miha Cimperman (JSI), Luka Stopar (JSI), Matej Polzelnik (JSI)	Review of terms of content
0.6	2023-12-21	Miha Cimperman (JSI), Luka Stopar (JSI), Matej Polzelnik (JSI)	First draft
0.9	2024-01-24	Matej Polzelnik (JSI)	Corrections following reviewers' feedback
0.10	2024-01-25	Matej Polzelnik (JSI)	Final draft
1.0	2024-01-29	Flavien Massi (INTRA)	Final version

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 EXECUTIVE SUMMARY

The primary goal of the CONDUCTOR project is to create, integrate, and showcase cutting-edge, sophisticated traffic and fleet management solutions to enhance the efficient and optimal transportation of passengers and goods. To achieve this objective, advanced Machine Learning (ML) and Artificial Intelligence (AI) technologies will be applied, and the resultant technologies will be developed and integrated and validated through three specific use cases (UCs). The work presented in this report focuses on Tasks 3.1 and 3.2 and aims to present the data harmonization and data fusion concepts design and implementation of the CONDUCTOR project.

The main objective of the deliverable was to develop methodologies related to data harmonization and data fusion process, including: (1) common data model design, (2) data space architecture, (3) big data architecture (4) data fusion methodologies for five different implementation scenarios.

FIWARE's smart data models were chosen as the foundation for harmonization within CONDUCTOR, aligning with the project goals. Chapter 3 highlights the utilization of a common data model to structure the Context Broker, managing the entire context information lifecycle. This includes updates, queries, registrations, and subscriptions, fostering semantic-level data integration and management. The adoption of common information models, coupled with data space design and big data architecture deployment, ensures seamless application integration and facilitates efficient exploration of CCAM services. Harmonized data models enhance sharing and exchange of information among project components.

Five significant developments were identified in the context of CONDUCTOR's data fusion tasks, each relevant for designing new traffic management strategies. These include characterizing delivery trips and estimating delivery demand, identifying unusual traffic patterns during large-scale events, creating a framework for smartphone-based data analytics, specifying FleetPy-Aimsun coupling, and developing space-time context and heterogeneous data fusion. The report provides detailed methodologies and initial implementation steps for each, to be applied and refined in the diverse use cases as the project progresses.

Although each development is initially framed within a specific UC, a "from-particular-to-general" approach is employed during definition and implementation. This allows for the formulation of general methodologies applicable across multiple UCs, enhancing flexibility and scalability. The report offers an initial version of methods, designs, and specifications for final data integration, subject to updates, testing, and validation in the designated UCs and pilot setups. The refined versions will be reported in the final reporting phase.

Keywords: Common Information Model, CCAM Data Space, Data harmonization, Data Fusion

2 INTRODUCTION

This report, as a result of Tasks 3.1 and 3.2 of the project, presents the data harmonization and data fusion concepts design and implementation, that are being developed in the context of the CONDUCTOR project and will be used in the use cases. These objectives are clustered into two main groups:

1. Data harmonization objectives:

- O1.1.: To design and develop a framework for reference data model implementation. The important part of the objective is to use the domain-specific smart data models, representing standardized domain uniform data models, supported by FIWARE, IUDX, TM Forum, OASC and others. In the process, the available smart data models are investigated and assessed for their suitability for representing data entities within CONDUCTOR. The basic methodology for the mapping of data entities should be represented and specified with Proof of Concept (PoC) in the final integration implementation.
- O1.2: To design and implement big data architecture for data space, to allow the efficient execution of machine learning algorithms and optimisation models for time-critical tasks. The design should include descriptions of the fundamental principles and structures underlying the big data architecture, including parallelization and distributed data processing and platform configuration. The design specifications should also provide an initial test of the main deployment concepts, such as container deployment on Kubernetes cluster, etc. The big data architecture specifications should provide conceptual schemas, selection and description of technologies and description of implementation scenario and configurations.
- O1.3.: To design CCAM and traffic data space architecture that will enable: various data integration patterns (pub/sub), data transformation (maintaining common data model for the domain of CONDUCTOR project), data management procedures (such as Extract Transform Load, ETL) for various data categories and data security/data access mechanisms at various levels. For this purpose, specifications for the “Context Broker” are needed accompanied with appropriate technologies that will be selected, to enable the management of complex data pipelines and the common data model.

2. Data Fusion Objectives:

- O2.1.: To develop a data fusion methodology for characterization of last-mile parcel delivery trips and estimation of delivery demand from a wide variety of data sources (mobile network data (MND), e-commerce survey data, delivery data, etc.). The development allows the characterisation of the e-commerce users by their place of residence and different socio-demographic features (such as age and gender), as well as the identification of the last-mile delivery trips based on mobility patterns extracted from MND. This identification is used to provide more detailed transport demand information segmented by mode and to develop coordination strategies between last-mile delivery and demand-responsive transport (DRT) for the Urban logistics UC (UC3) of the project.
- O2.2.: To develop a framework for identifying unusual traffic patterns caused by large-scale events. Taking advantage of sensors and social networking platforms, unusual traffic patterns can be detected, and their development can be traced in real-time. To this end, data fusion, statistical learning and ML techniques are combined to fuse and extract new meaningful features from different data sources and predict traffic events. The information generated by

the tool can provide first responders with the right information to monitor traffic conditions and create situational awareness, supporting the decision-making process.

- O2.3.: To develop a framework for actionable smartphone-based data analytics. The framework will establish a generic detailed modelling plan to address issues of data processing and analysis for stream data coming from smartphone sensors, that creates actionable information out of raw data. This plan is based on five main steps: sensing, data processing, data modelling, model exploitation, and adaptation. The information obtained with this process can be used for driving analytics, mobility analytics, and parking analytics.
- O2.4.: To develop a methodology for FleetPy—Aimsun coupling. FleetPy, a Python-based DRT simulation tool, does not have an integrated traffic micro-simulation functionality. The aim of this coupling is to fill in this gap to improve FleetPy simulation potential by using the capabilities of the Aimsun Next simulation tool. The bridge allows the consideration of a more realistic traffic simulation in the FleetPy control decisions which replicates the unexpected delays that DRT might face in real traffic.
- O2.5.: To design space-time context for heterogeneous data. The purpose of the objective is to develop a methodology and implementation design for the embedding of heterogeneous data into a graph representation, that enables augmentation and contextualization of data source beyond basic single data representation. More importantly, the specifications address the methods for feature vectors embedding into common space-time context representation. The final specifications should include methods for feature extraction and data retrieval from space-time context graph.

The objectives represented are a structured description of main knowledge, functionalities and capabilities to be targeted at and were used as guidelines for research and development work under T3.1. and T3.2. The results are represented in the Chapter 3 and Chapter 4.

2.1 Outline of the Deliverable

This deliverable begins with an executive summary, offering a brief overview of the CONDUCTOR project's objectives, goals, and a description of the deliverable's contents. Progressing to the introduction chapter, the background establishes the context by explaining overarching challenges. The subsequent section on objectives and contributions defines the document's purpose and potential impact, followed by a detailed outline of the deliverable's structure and content.

The main content of the deliverable is consolidated under two chapter, namely: (a) Chapter 3 Data space design and harmonization, (b) Chapter 4 Data Fusion.

Chapter 3 describes the data harmonization part of CONDUCTOR project, with the architecture of the data space and the introduction of the “context broker” as the main agent for managing data process flow. The basic architecture of processing heterogeneous data is represented and methods for semantic data harmonization are introduced, including an example of harmonization of three basic data sources as a Proof of Concept (PoC). In the continuation, the section presents the big data architecture and the appropriate technologies to implement the designed data space concept, including configurations and final testing with ML models deployment capabilities. The architecture presented includes the complete deployment pipeline, including continuous integration and continuous development mechanisms. The big data architecture is finally benchmarked for performance with three basic ML algorithms, offering results on performance evaluation.

Chapter 4 presents the data fusion algorithms needed for the development of the CONDUCTOR decision support models and tools. The chapter describes data fusion methods and implementation in five main strategies that are employed in CONDUCTOR, namely: (a) Characterisation of delivery trips and estimation of delivery demand from mobile network, surveys and logistic operation data, (b) Identification of unusual traffic patterns caused by large-scale events, (c) Framework for actionable smartphone-based data analytics, (d) FleetPy—Aimsun coupling specification, (e) Space-time context and heterogeneous data fusion. Each of the strategies is described in the context of UC implementation, including definition of problem, data portfolio used and data fusion methods applied. The data fusion methods were applied, tested and evaluated to provide estimations for final implementation.

The document concludes with summarization of key challenges, important findings and expected results in further development for both: data harmonization and data fusion part of the deliverable.

3 DATA SPACE DESIGN AND HARMONIZATION

This chapter outlines the efforts made to establish the CONDUCTOR data space, which will serve as the data architecture for the project's models and solutions. The objective of the data space is to provide a comprehensive solution that encompasses data sovereignty and control, interoperability, and trustworthiness. To meet the requirements of the CONDUCTOR project, a dual approach has been identified:

1. Leveraging tools from the Fiware ecosystem, a publish/subscribe framework is being implemented to unify data providers and consumers. This approach will primarily be utilized for open data and real-time data streaming applications.

2. IDSA connectors are being realised for the transfer of data, enabling secure and effective communication and exchange in the developed data space. Note, that for the purposes of CONDUCTOR the Minimum Viable Dataspace (MVD) is being implemented, with additional features to be added in the future if necessary.

The section is divided into three parts: (i) a description of the chosen data harmonisation approach, (ii) the specification of the proposed MVD architecture, and (iii) an explanation of the experimentation conducted to evaluate various big-data approaches for integration into the project's data architecture.

3.1 Data Harmonization

3.1.1 Concept and Approach

Data harmonisation entails the adoption of common information models for representing schemas and semantics for data to be used by applications. CONDUCTOR develops a variety of tools and algorithms to support the investigation of CCAM services and therefore a harmonised representation of data models is necessary to facilitate seamless integration of applications. A list of the identified data sources for the different CONDUCTOR use cases can be seen in Appendix A. As can be seen, data sources for the representation of transport supply and demand, as well as weather, air quality, land use and telecom data are among those available for the project.

Based on the available data sources, FIWARE's smart data models[1] were selected as the basis for harmonisation within CONDUCTOR. This specific initiative is a collaboration program, led by FIWARE, IUDX, TM Forum, OASC and others. The aim is the adoption of a reference architecture and uniform data models for interoperable and integrated smart solutions and systems. The use of FIWARE's smart data models for data harmonisation in CONDUCTOR is based on their suitability and alignment with the project's goals. By adopting common information models for data representation, CONDUCTOR can ensure seamless integration of applications and enable efficient investigation of CCAM services. The harmonised representation of data models allows easier sharing and exchange of information among different components of the project.

[1] https://www.fiware.org/smart-data-models/#:~:text=A%20smart%20data%20model%20includes,examples%20of%20the%20payloads%20for

3.1.2 Data Harmonisation Developments

During the initial phase of the project, we conducted an investigation into the available smart data models and assessed their suitability for representing data entities within CONDUCTOR. In this regard, we identified the following data models as appropriate for the purpose of harmonization:

- ItemFlowObserved[2]: This data model can effectively capture readings obtained from infrastructure sensors such as loop detectors. These readings encompass various traffic engineering parameters including average speed, flow, occupancy, and others.
- WeatherObserved[3]: This data model represents observations of weather conditions at specific locations and times.

To demonstrate the harmonization process, we have selected the following CONDUCTOR data sources:

- Attica Traffic Data
- DARS Traffic Data
- weatherapi Weather Data

3.1.3 Attica Traffic Data

The mapping of the data entities for the harmonisation of the Attica region traffic data[4] can be seen in Figure 1.

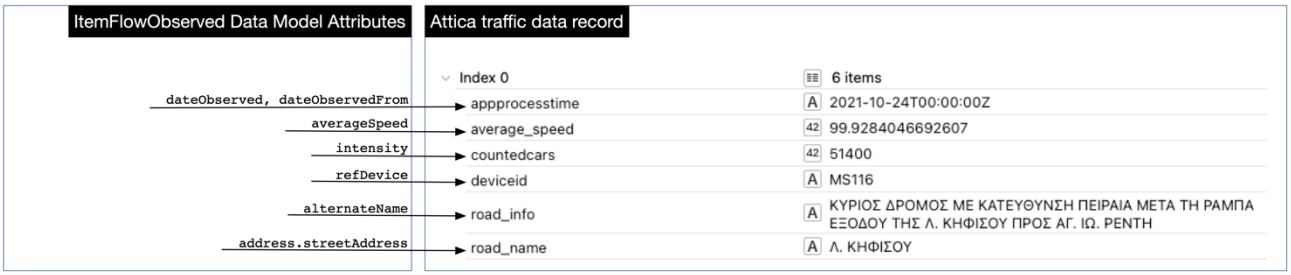

Figure 1 Data entities mapping for Attica region traffic data

The corresponding data model entity record in JSON format following the NGSI v2 information model can be seen below.

```
{
    "id": "FlowObserved:Attica_MS116",
    "type": "ItemFlowObserved",
    "address": {
      "addressCountry": "GR",
      "addressLocality": "ΑΤΤΙΚΗ",
      "streetAddress": "Λ. ΚΗΦΙΣΟΥ"
    },
    "alternateName": "ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΠΕΙΡΑΙΑ ΜΕΤΑ ΤΗ ΡΑΜΠΑ
ΕΞΟΔΟΥ ΤΗΣ Λ. ΚΗΦΙΣΟΥ ΠΡΟΣ ΑΓ. ΙΩ. ΡΕΝΤΗ",
```

[2] https://github.com/smart-data-models/dataModel.Transportation/tree/master/ItemFlowObserved

[3] https://github.com/smart-data-models/dataModel.Weather/blob/master/WeatherObserved/doc/spec.md

[4] https://www.data.gov.gr/datasets/road_traffic_attica/

```
    "averageSpeed": 99.9284046692607,
    "dateObserved": "2021-10-24T00:00:00Z",
    "dateObservedFrom": "2021-10-24T00:00:00Z",
    "dateObservedTo": "2021-10-25T00:00:00Z",
    "intensity": 51400,
    "refDevice": "MS116"
}
```


3.1.4 DARS Traffic Data

The mapping of the data entities for the harmonisation of the DARS traffic data[5] (Slovenian road network) can be seen in Figure 2.

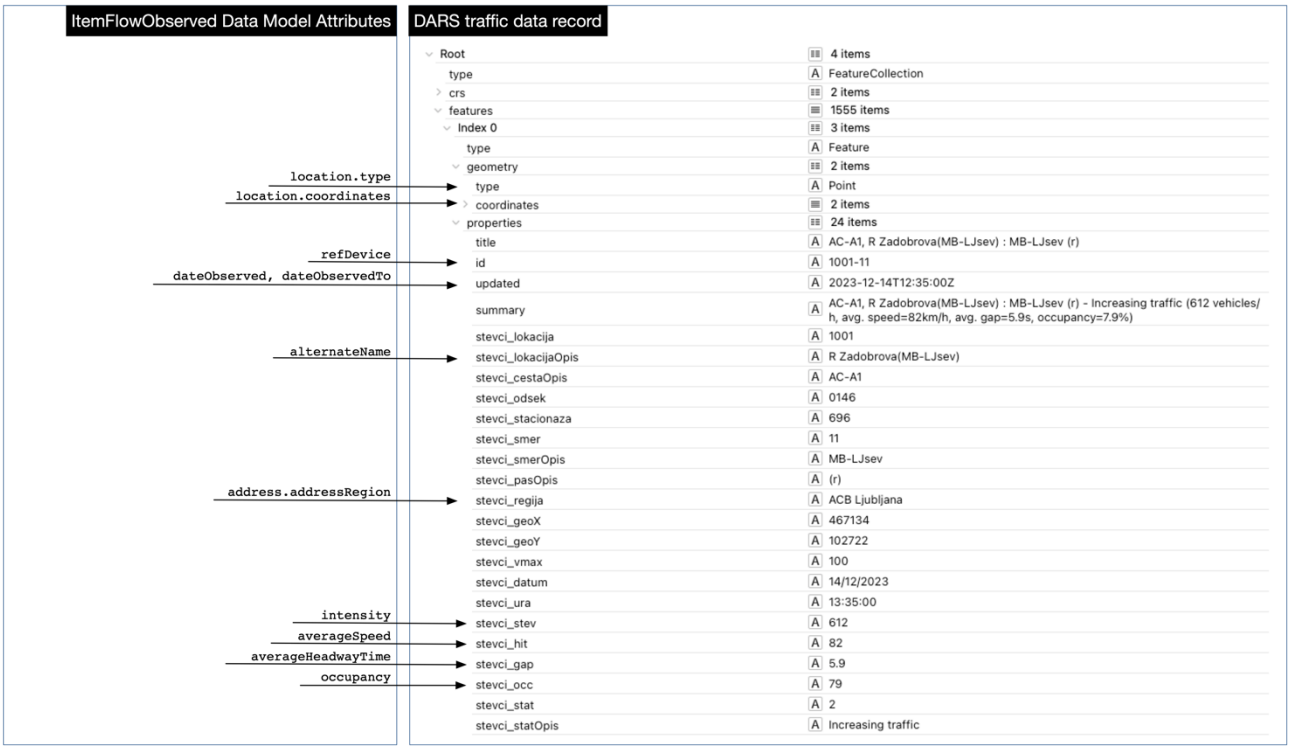

Figure 2 Data entities mapping for DARS traffic data

Similarly, the corresponding data model entity record in JSON format following the NGSI v2 information model can be seen below.


```
{
    "id": "FlowObserved:DARS_1001-11",
    "type": "ItemFlowObserved",
    "address": {
      "addressCountry": "SI",
      "addressRegion": "ACB Ljubljana"
    },
    "location": {
        "coordinates": [
```


[5] https://www.dars.si

```
            14.570367,
            46.06786
        ],
        "type": "Point"
    },
    "alternateName": "R Zadobrova(MB-LJsev)",
    "averageSpeed": 82,
    "dateObserved": "2023-12-14T12:35:00Z",
    "dateObservedFrom": "2023-12-14T12:30:00Z",
    "dateObservedTo": "2023-12-14T12:35:00Z",
    "intensity": 612,
    "averageHeadwayTime": 5.9,
    "occupancy": 7.9,
    "refDevice": "1001-11"
}
```


3.1.5 weatherapi Weather Data

The mapping of the data entities for the harmonisation of the weather data from the weatherapi.com service can be seen in Figure 3.

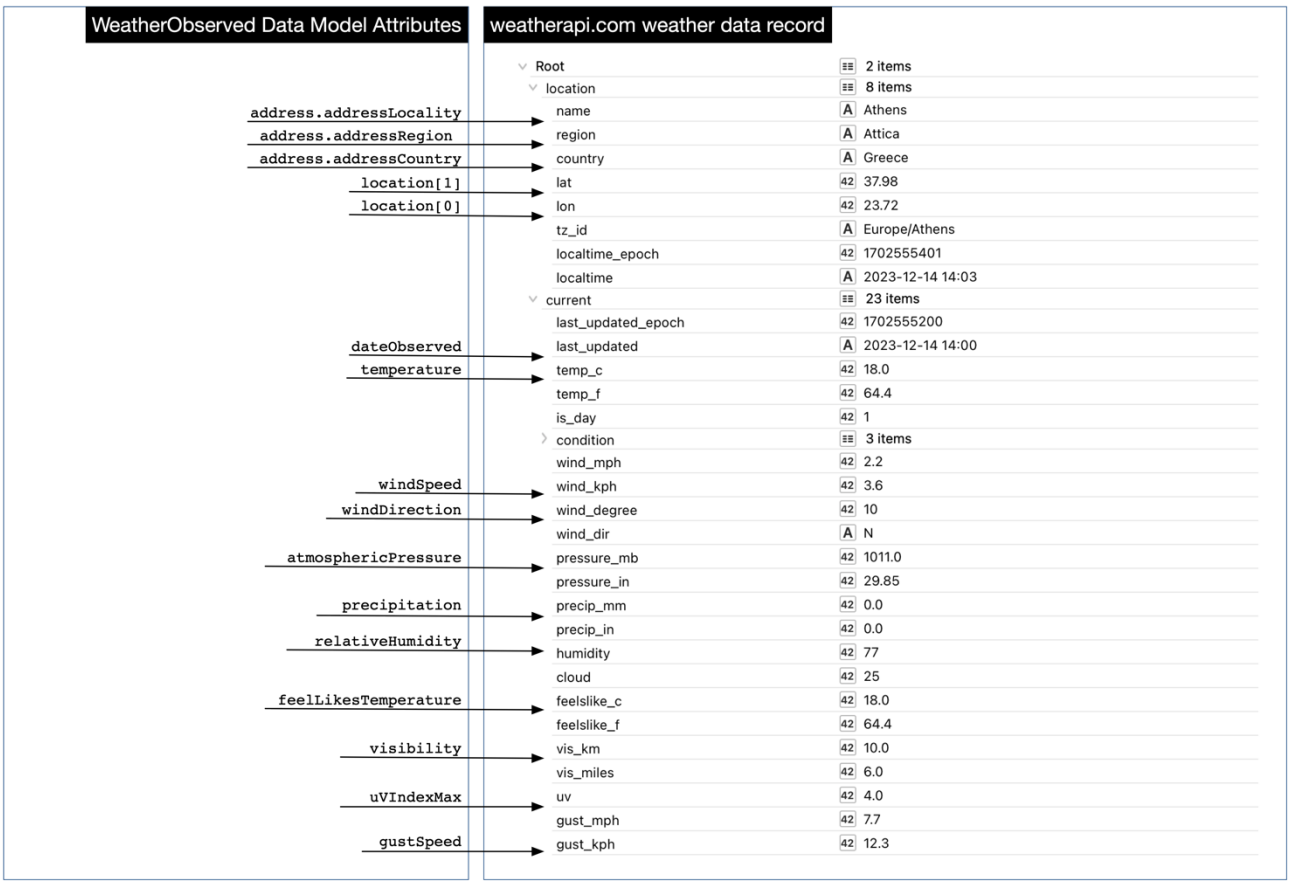

Figure 3 Data entities mapping for the weatherapi weather data

The corresponding data model entity record in JSON format following the NGSI v2 information model can be seen below.


```
{
    "id": "Greece-WeatherObserved-Athens-2023-12-14T14:03:00+02:00",
    "type": "WeatherObserved",
```

```
    "address": {
      "addressLocality": "Athens",
      "addressRegion": "Athens",
      "addressCountry": "GR"
    },
    "dateObserved": "2023-12-14T14:03:00+02:00",
    "location": {
      "type": "Point",
      "coordinates": [
        23.72,
        37.98
      ]
    },
    "atmosphericPressure": 1011.0,
    "windSpeed": 3.6,
    "windDirection": 10,
    "temperature": 18.0,
    "feelLikesTemperature": 18.0,
    "precipitation": 0.0,
    "uVIndexMax": 4.0,
    "relativeHumidity": 77,
    "visibility": "excellent",
    "gustSpeed": 12.3}
```


3.2 CONDUCTOR Minimum Viable Dataspace

3.2.1 Minimum Viable Dataspace Architecture

The overall architecture of CONDUCTOR's MVD can be seen in Figure 4 below.

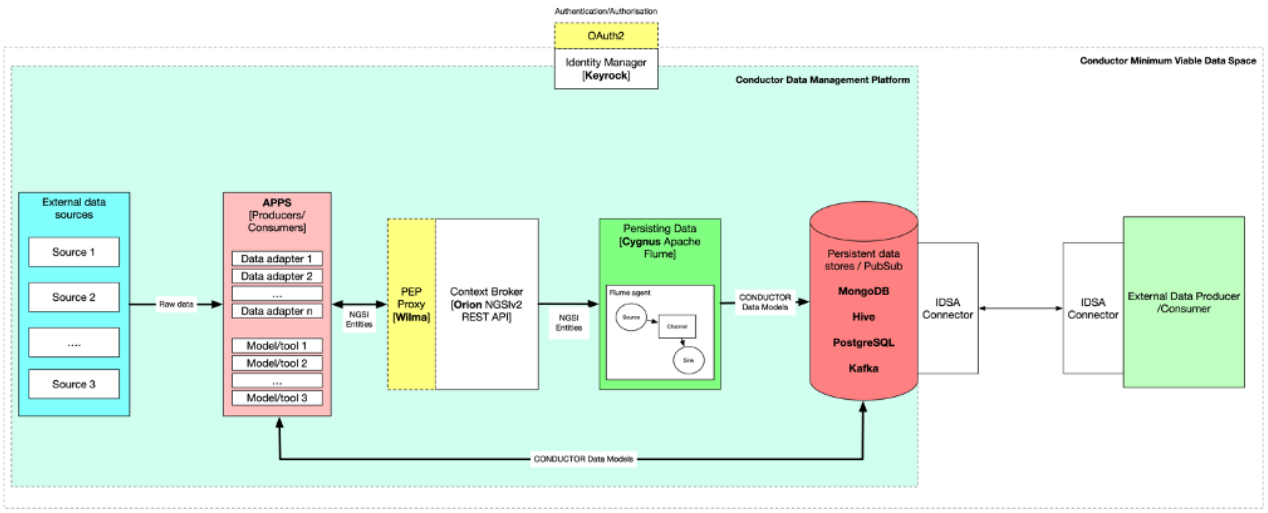

Figure 4 MVD Architecture

The overall solution is composed of two components, (i) the data management platform, which realises a Fireware compliant context management framework, (ii) IDSA data adaptors, which facilitate secure transfer of data between entities based on established agreements (contracts) between the interested parties.

3.2.2 Data Management Platform

At the heart of the approach resides the ORION context broker, which offers holistic management of context information (data entities) including insertion, updating, querying and deletion. In addition, the context broker facilitates a publish/subscribe paradigm where systems can receive context information as soon as it is published on the broker. For the purposes of CONDUCTOR an example of the interaction of different systems with the context broker can be visualised in Figure 5. ORION allows the issuing of different commands (using conventional REST API commands) for managing context information. These are briefly described below:

- POST: for creating a new entity on the broker. The payload can include the data model entity records shown in sections 3.1.3 - 3.1.5.
- PUT: can be used for replacing all the attributes of a given entity, removing the previously existing ones. The payload includes a list of the new attributes.
- PATCH: can be used for updating the value of an entity's attribute if the attribute already exists. The payload includes the attributes that need to be updated.
- DELETE: can be used for deleting an entity.
- GET: can be used for retrieving context from the broker.

The first four commands are being utilised by context producers, while the last command is primarily being adopted by context consumers.

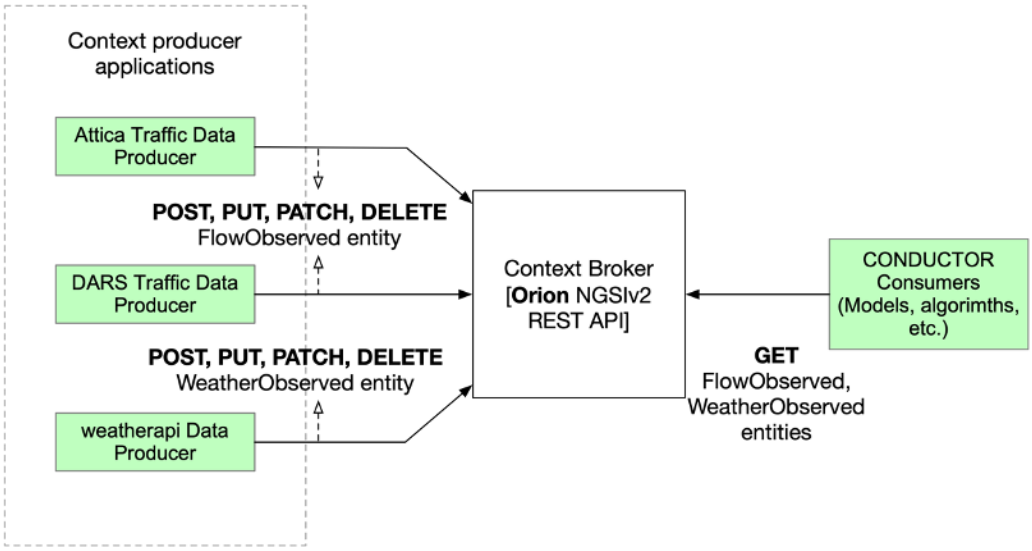

Figure 5 ORION context broker interactions

The remaining elements of the architecture are briefly described below:

- **External Data Sources:** These sources provide raw data that serves as the foundation for CONDUCTOR's insights and decision-making processes. As stated earlier in the chapter, the data sources available to the project can be found in Appendix A.
- **APPS (Producers/Consumers):** Applications within the CONDUCTOR ecosystem act as both producers and consumers of data. They are responsible for generating valuable insights and utilizing information from other sources. The developed adaptors will support the smart data models described in section 3.1.1 above. More information regarding the adopted publish/subscribe approach is presented in section 3.2.2 below.

- **Authentication and Authorisation:** For the realisation of secure access to the resources managed by the data space, FIWARE's **Keyrock**[6] and **Wilma**[7] components will be utilised. Keyrock will provide identity management functionalities, allowing APPS developers to register their solutions before publishing or accessing protected data. Wilma is a Policy Enforcement Point (PEP) proxy that will act as intermediary between the developed APPS and the Context Broker, adding an additional protection layer to the architecture. These components will be setup during the next phase of the project and once the CONDUCTOR's IT infrastructure is fully operational.
- **Persisting Data [Cygnus]: Cygnus**[8] is a FIWARE component based upon Apache's Flume Source=>Channel=>Sink paradigm and is responsible for persisting data, ensuring its durability and availability for further analysis. The process involves a source, a channel, and a sink, forming a comprehensive data pipeline. As in the case of Keyrock and Wilma, Cygnus will be deployed on the project's IT infrastructure to offer data persistency for the context data produced to the Orion Context Broker.
- **Persistent Data Stores:** Cygnus offers data persistence to various third-party tools such as MongoDB, Hive, PostgreSQL, and Kafka. For the purposes of CONDUCTOR **MongoDB**[9] will form the primary persistent storage medium, while requirements for additional tools will be investigated in the second phase of the project.

3.2.3 Publish/Subscribe Model

In addition to the 'pull' interface (using GET commands), data consumers can retrieve data from the broker by 'subscribing[10]' to specific entities. This can be achieved through a subscription request that defines the entities of interest. Once a subscription has been verified, the subscribed application will receive asynchronous notifications every time the relevant entities have been updated. The following subscription request examples demonstrate how an application can receive data from a single traffic sensor, or from all traffic sensors available.

Payload of request for subscribing to data entities related to sensor with id FlowObserved:Attica_MS116.

```
{
    "description": "A subscription to get traffic data from sensor with
id FlowObserved:Attica_MS116",
    "subject": {
      "entities": [
        {
          "id": "FlowObserved:Attica_MS116"
        }
      ],
      "condition": {
```

[6] https://fiware-idm.readthedocs.io/en/latest/

[7] https://github.com/ging/fiware-pep-proxy

[8] https://fiware-cygnus.readthedocs.io/en/latest/

[9] https://www.mongodb.com

[10] https://fiware-orion.readthedocs.io/en/master/user/walkthrough_apiv2.html

```
        "attrs": [
        ]
      }
    },
    "notification": {
      "http": {
        "url": "http://localhost:8080/traffic"
      },
      "attrs": [
      ]
    },
    "expires": "2030-12-31T00:00:00.00Z"
  }
```

Payload of request for subscribing to data entities related to all traffic data sensors available.

```
{
    "description": "A subscription to get traffic data from all sensors
available",
    "subject": {
      "entities": [
        {
          "type": "ItemFlowObserved"
        }
      ],
      "condition": {
        "attrs": [
        ]
      }
    },
    "notification": {
      "http": {
        "url": "http://localhost:8080/traffic"
      },
      "attrs": [
      ]
    },
    "expires": "2030-12-31T00:00:00.00Z"
  }
```

3.2.4 IDSA Connectors

In the architecture of the CONDUCTOR data space, a crucial component is the implementation of IDSA connectors for secure and efficient data exchange. These connectors, specifically the data connectors sourced from the Eclipse open github repository[11] play a pivotal role in our system.

The Eclipse data connectors are designed to be compliant with the rigorous standards set by IDSA for secure data exchange. These connectors facilitate the integration of disparate data sources by providing a uniform interface for data transfer, ensuring interoperability and compliance with data

[11] https://github.com/eclipse-edc/MinimumViableDataspace/tree/main

sovereignty principles. This is particularly significant in environments where diverse data sets are involved, as it ensures seamless and secure data flow across different platforms and systems.

Our application of these connectors in the CONDUCTOR data space is aimed at enhancing the data exchange processes, aligning with the broader objectives of the project. By leveraging the capabilities of these connectors, we are able to ensure that data exchange within our architecture is not only efficient but also aligns with the high standards of security, compliance, and interoperability advocated by IDSA. The implementation of these connectors signifies a step forward in achieving a robust and scalable data architecture that can support the complex demands of the CONDUCTOR project.

In addition to implementing IDSA connectors from Eclipse in our architecture, we have taken a step further by containerizing these connectors, so they can be deployed through Docker. This approach involves encapsulating the connectors within containers, thereby enhancing their scalability, portability, and ease of deployment across different computing environments. Containerization also contributes to a more modular architecture, allowing for greater flexibility and efficiency in managing and updating the connectors.

To fully explore the potential of these connectors, we have conducted mock data exchanges within our data space. These exercises are designed to simulate real-world data exchange scenarios, allowing us to assess the performance, reliability, and security of the connectors in a controlled environment. By running these mock exchanges, we gained valuable insights into how the connectors handle various types of data, respond to different load conditions, and integrate with other components of our data architecture. This hands-on experimentation is crucial for fine-tuning the connectors and ensuring that they meet the specific needs and challenges of the CONDUCTOR project.

3.3 Big Data Architecture and Design

The big-data architecture within the CONDUCTOR project aims to provide a framework of tools which will allow the efficient execution of machine learning algorithms and optimisation models for time-critical tasks. This section outlines the fundamental principles and structures underlying the design of the big data architecture, which serves as the backbone for data processing and analytics in the context of fleet and traffic management.

3.3.1 Architecture Overview

The foundation of the CONDUCTOR project's data processing lies in a container-based infrastructure as described in D4.1. Deploying and scaling containerized applications ensures flexibility and efficient resource management for seamless data processing across the project (Figure 6).

- **Container-based Infrastructure**: Serving as the cornerstone of the project's data processing, a container-based architecture provides a scalable and flexible infrastructure, which can be orchestrated through the deployment and management of containerized applications for adaptability to varying workloads.
- **Containerization Strategy**: Containerization plays a crucial role in maintaining consistency and isolating processes within the project. Containers enhance agility by encapsulating applications and their dependencies, enabling efficient deployment across diverse environments.
-

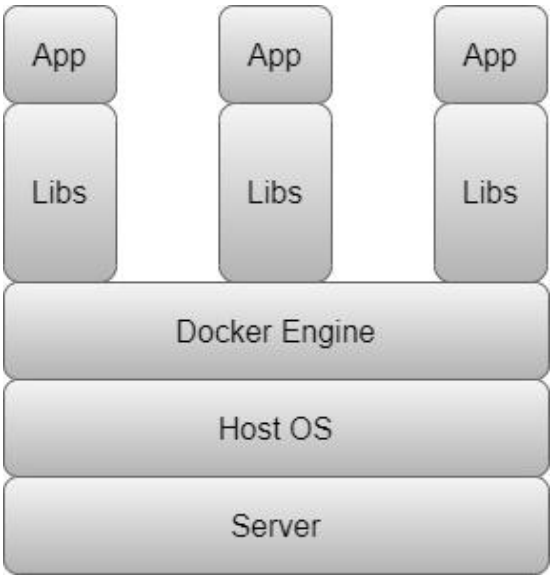

Figure 6 Representation of three apps running on three different containers

3.3.2 Distributed Data Processing with Kubernetes and Apache Spark

Kubernetes is increasingly being utilized for the development of container-based web applications on physical computers within Platform-as-a-Service (PaaS) clouds. It enables the scalability of applications through dynamic workload changes. Kubernetes follows a master-slave architecture, as depicted in Figure 7. The master node is responsible for managing the Kubernetes system and serves as the entry point for all administrative tasks. It coordinates the execution of tasks where the actual services are performed.

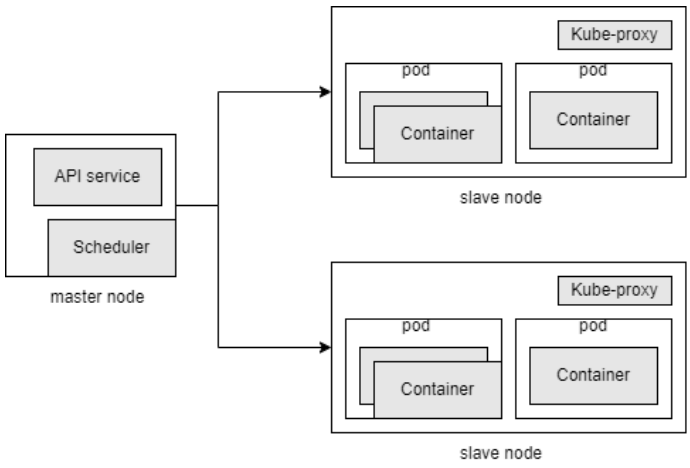

Figure 7 Kubernetes Architecture

In Kubernetes, pods, rather than containers, serve as the smallest units of computation that run on slave nodes. A pod can encapsulate one or multiple containers and is assigned a unique IP address. Each container within a pod shares the network namespace, including the IP address and network ports. Communication between pods running on different physical computers is facilitated by Kube-proxy, a component of Kubernetes. To maintain flexibility and reliability, pods are typically deployed with a CPU demand of one core or less, allowing for flexible deployment across various nodes.

Spark can also utilize Kubernetes as its cluster manager, similar to other administrators. In Kubernetes, all Spark drivers and executors run within pods and are scheduled by Kubernetes' native

scheduler. Upon submitting a Spark application to a Kubernetes cluster, a Spark driver is created and initially runs within a pod. The driver then creates Spark executors, which also run within pods, to connect to them and execute the application's code. Once the application is completed, the executor pods are terminated and cleaned, while the driver pod retains the log files and remains in a "completed" state in the Kubernetes API until it is eventually "garbage collected" or manually deleted (Zhu et al., 2020).

Central to the Big Data architecture is Apache Spark, a powerful tool for distributed data processing. Spark's capabilities enable handling large-scale datasets efficiently, enabling parallel processing and laying the groundwork for fusion and analysis within the project.

- It is chosen for its capability to handle large-scale datasets through distributed processing. Spark‘s architecture supports parallel computation, allowing for efficient and rapid analysis of diverse data sources.
- It seamlessly integrates with other CONDUCTOR components. It acts as the backbone for parallel processing, facilitating data fusion and analysis across various modules, ensuring a cohesive and interoperable ecosystem.

3.3.3 Deployment Configurations

This section describes the setups employed for our experimental endeavours in distributed computing utilizing Kubernetes and Apache Spark. These configurations encompass various infrastructure strategies, each possessing its own distinct attributes and trade-offs. The selection of these formats is of utmost importance as it significantly impacts the performance, scalability, and portability of Apache Spark applications, and thus the overall CONDUCTOR big-data architecture. Our experimental work encompasses three distinct configurations, each representing a different technological layer. These are:

- Single Spark Cluster node
- Two Spark Cluster nodes (Figure 8)
- Four Spark Cluster nodes (Figure 9)

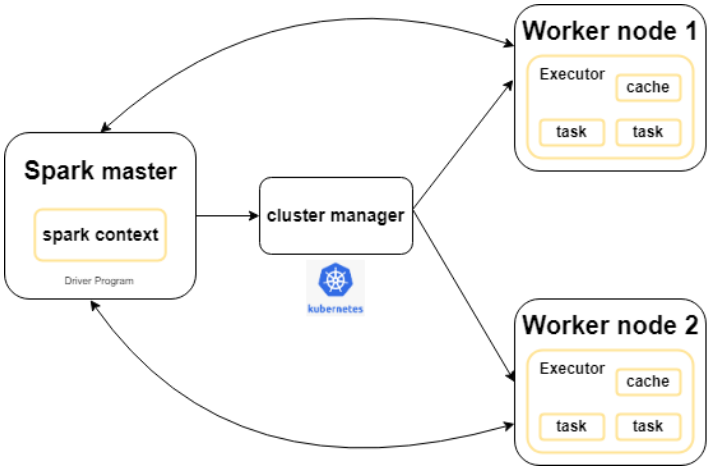

Figure 8 Two node spark cluster: Execution flow

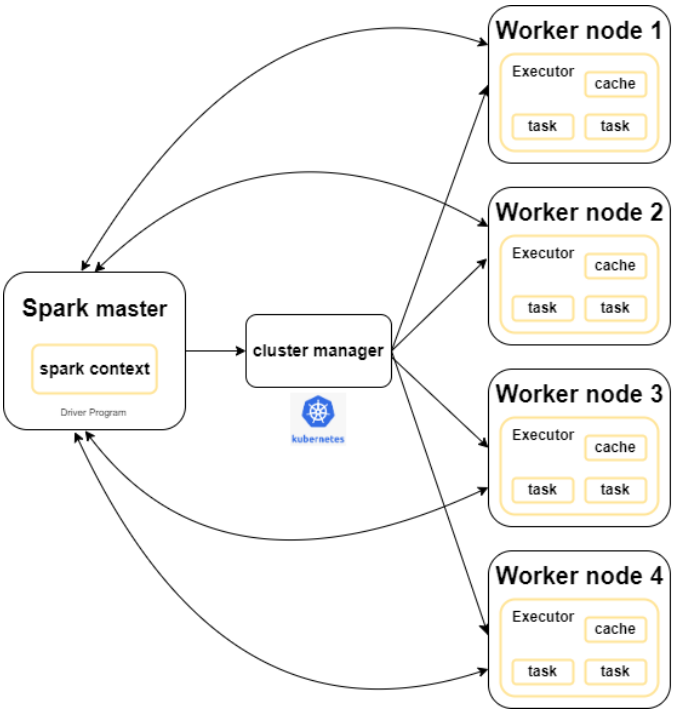

Figure 9 Four node spark cluster: Execution flow

3.3.4 Experimental Results

In this section, we will thoroughly examine the selected machine learning algorithms and their implementation within the Apache Spark framework. These algorithms have been chosen based on their relevance to identifying phenomena and their diversity in the field of machine learning. From many available, we selected the following algorithms: Multiclass Logistic Regression, Decision Tree, and Naive Bayes. The rationale behind this selection is to facilitate a comprehensive comparison of different configurations, taking into consideration the unique characteristics of each algorithm.

Multiclass Logistic Regression: Multiclass logistic regression is used when the prediction of more than two classes is necessary. In multiclass logistic regression, the goal is to predict the probability of an input belonging to each class and this machine learning model is suitable for incident detection applications. Apache Spark offers a flexible implementation of Multiclass Logistic Regression through the LogisticRegression class. This algorithm is known for its simplicity and interoperability, which can be crucial in understanding incident detection results.

Decision Tree: The Decision Tree algorithm is a classic machine learning algorithm that creates a tree-like model of decisions and their possible connections. It is widely used in various domains, including incident detection. Apache Spark provides an implementation of the Decision Tree Classifier algorithm. Decision Trees are known for their diversity and ease of interpretation, which can be crucial in comprehending detection results.

Naive Bayes Classifier: The Naive Bayes algorithm is a probabilistic algorithm based on Bayes' theorem. It is particularly useful for text classification and categorical data, but it can also be applied to incident detection tasks that involve categorical outputs (i.e. vehicle collision, vehicle breakdown, etc.). Apache Spark enables the implementation of Naive Bayes through the NaiveBayes class. Despite its simplicity and the assumption of feature independence, Naive Bayes often demonstrates impressive performance in practice and can be efficient for handling large data sets.

The results from the execution of the algorithms can be seen at the table below (Table 1). As it can be seen the performance of the algorithms improves with the increase of the cores used as part of the architecture.

Table 1 Big-data architecture performance experimental results

Algorithms	Configuration / Setup	Performance (KPIs)		
		Time	%Cpu (us)	Memory
Multiclass Logistic regression	One node spark cluster	137.73 s	55.9	438.67 MiB
	Two node spark cluster	88.40 s	55.4	259.61 MiB
	Four node spark cluster	62.89 s	52.4	169.90 MiB
Decision tree	One node spark cluster	549.47 s	61.9	1011.34 MiB
	Two node spark cluster	344.128 s	64.4	598.15 MiB
	Four node spark cluster	249.46 s	61.1	391.46 MiB
Naive Bayes	One node spark cluster	415.95 s	88.5	761.92 MiB
	Two node spark cluster	263.25 s	84.5	451.05 MiB
	Four node spark cluster	183.96 s	90.8	295.20 MiB

4 DATA FUSION

This section presents the data fusion algorithms needed for the development of the CONDUCTOR decision support models and tools. The objective of these algorithms is to transform harmonised data (filled by different sources) into medium and high-level features to be used by the decision support models, including, among other, pattern discovery from heterogeneous data and predictive modelling methods for mapping among data of various granularities. The developed algorithms will be used either to feed the development of new mobility and traffic models or to represent observed behaviours to be modelled in the CONDUCTOR UCs.

4.1 Concept and Approach

The fusion of diverse data mobility-related information allows the reconstruction of traffic and mobility patterns, essential in crafting effective traffic and fleet management strategies. This fusion involves harnessing data from a set of sources, such as GPS devices, mobile phone records, surveys, etc. Each of these sources contributes complementary information, enriching the overall understanding of mobility dynamics.

The reconstructed patterns serve as a foundational basis for designing intelligent traffic and fleet management strategies. By leveraging insights gained from the analysis of varied data sets, transportation systems can be optimized, routes can be streamlined, and congestion mitigated.

It is noteworthy that different combinations of data sources yield equivalent patterns and mobility indicators, highlighting the flexibility and adaptability of this approach.

In the context of CONDUCTOR, the following data fusion developments and analysis have been identified by Nommon, INTRA, NTUA, TUM, and JSI, respectively, as relevant for the design of new traffic management strategies:

- Characterisation of delivery trips and estimation of delivery demand from mobile network, surveys and logistics operation data.
- Identification of unusual traffic patterns caused by large-scale events.
- Framework for actionable smartphone-based data analytics.
- FleetPy—Aimsun coupling specification.
- Space-time context and heterogeneous data fusion.

Each development is framed within one of the CONDUCTOR UCs. However, even though the data fusion algorithms have been identified based on the UCs needs, during the definition and implementation phases, a from-particular-to-general approach is being followed, in which each development allows the definition of general methodologies that can be extrapolated, whenever other data sources with similar characteristics are available.

4.2 Data Fusion Developments

Next, the data fusion developments are described in detail.

4.2.1 Characterisation of Delivery Trips and Estimation of Delivery Demand

4.2.1.1 Introduction

Nommon is developing an algorithm for the identification, characterisation and prediction of parcel delivery demand to be used in the Urban logistics use case (UC3) of the project. This UC investigates solutions aimed at the optimal integration of urban freight distribution with DRT, in order to reduce last-mile parcel delivery-related traffic. The goal is to leverage the excess capacity of DRT vehicles during periods of lower demand for the last-mile delivery of freights compatible with passenger transport. For that, both the parcel delivery demand and the DRT demand need to be estimated, in order to identify the valley hours in the DRT demand and coordinate both services to define optimal routes in those periods.

This algorithm is being implemented in two phases, each providing an incremental level of detail:

- phase 1: estimation of the delivery demand from surveys. In this phase, the delivery demand data provided by the Spanish National Statistics Institute aggregated at Spanish province level is disaggregated into smaller administrative levels, such as district or census tract.
- phase 2: identification and characterisation of delivery trips. In this phase, a longitudinal behavioural analysis on mobility patterns extracted from MND, enriched with e-commerce delivery data, is performed to identify the delivery trips and characterise the delivery flows.

Next, the data needed and the methodology for the implementation of each phase is described.

4.2.1.2 Data Used

The data needed for the identification of delivery trips and prediction of delivery demand are the following (see Deliverable D1.2 (CONDUCTOR Consortium, 2023) for more details on the data sources):

- data needed for phase 1:
  - Spanish census data, provided by the Spanish National Statistics Institute (INE). This dataset contains population data, characterised by age group and gender at Spain census tract level.
  - Survey on equipment and use of information and communication technologies in households, provided by the INE. This survey contains information about the general use of information and communication technologies, in particular, about the use of e-commerce for private reasons of the Spanish residents, characterised by purpose and sociodemographic characteristics of the population.
- data needed for phase 2:
  - MND, provided by one of the largest telecom companies in Spain. This data source contains mobile phone Call Detail Records (CDRs) and probes data.
  - E-commerce delivery data, provided by Citylogin. Citylogin is a last-mile delivery logistic company located in Madrid that has shared their delivery data, previously anonymised, with Nommon under private agreement for the context of this project. The data provided include information of goods travel demand, including main delivery stops and delivery itineraries.
  - land use and points of interest (POIs). The land use information is provided by the Spanish National Geographic Information Centre and contains geometry of each land use

type per Autonomous Community of Spain. The points of interest data are generated by Nommon for this development based on the location of logistic and delivery hubs.

4.2.1.3 Methodology for Phase 1

The objective of phase 1 is to provide information about the volume of e-commerce demand generated in a region based on its sociodemographic characteristics, i.e., its population distribution by gender, age, income level, household size, etc. For that, Nommon has defined a methodology to disaggregate the available e-commerce delivery demand of a large region to smaller subregions. This methodology is based on the hypothesis that: i) sociodemographic characteristics of the population are good explanatory variables of the delivery demand, and ii) that the relation between the sociodemographic characteristics and the observed demand patterns in big regions is preserved in smaller subregions, i.e., people from the same sociodemographic group behave similarly regardless of their place of residence (within the same region).

The methodology defined is depicted in Figure 10 and consist in the following steps:

1. Delivery demand distribution per each sociodemographic group, defined as a combination of sociodemographic characteristic (age, gender, income level, or household size), is extracted from the analysis of the e-commerce demand (delivery demand) provided by the survey on equipment and use of information and communication technologies in households.
2. Delivery demand of each sociodemographic group is distributed to each subregion according to the distribution of the different sociodemographic groups in each subregion. This distribution is carried out in the following manner:
    a. the percentage of each group living in each subregion is computed, creating a probability distribution of the delivery demand in each subregion.
    b. the probability distribution is applied to the buyers of each group one by one (to preserve an integer number of buyers) to assign them to a subregion.
    c. the probability distribution is updated each time a buyer is assigned to a subregion by subtracting the buyer from the population of the sociodemographic group it belongs to, both at region and subregion levels. This ensures that the process does not assign more buyers than available population in each group to each region.

To illustrate this dynamic probabilistic assignment, let us suppose that buyer b belongs to sociodemographic group g, the total population of that group in the region is p_g, and there are n subregions, $r_1, \dots, r_n$, with population of that group $d_1, \dots, d_n$, respectively. The probability distribution of group g before the assignment is:

$$\text{prob}(g) = \left(\frac{d_1}{p_g}, \dots, \frac{d_n}{p_g}\right).$$

Now, let us assume that buyer b is assigned to the subregion r_k, then, the updated probability distribution of group g (after the assignment) is:

$$\widehat{\text{prob}}(g) = \left(\frac{d_1}{p_g - 1}, \dots, \frac{d_k - 1}{p_g - 1}, \dots, \frac{d_n}{p_g - 1}\right).$$

This process yields a disaggregation of the delivery demand in each subregion considered and a characterisation of the delivery demand in terms of the sociodemographic characteristics of the population.

This methodology is being tested and validated in the Madrid region. Nevertheless, with the from-particular-to-general philosophy in mind, this methodology can be applied to any region in which demand delivery information characterised per sociodemographic group is available, as long as both

the considered region and the sociodemographic characteristics meet the starting hypothesis. Furthermore, it is not limited to the context of delivery demand, but can be applied to any case in which it is required to disaggregate information provided at a regional level that depends on the sociodemographic characteristics of its population (or any other kinds of characteristics that meets the starting hypothesis in the region).

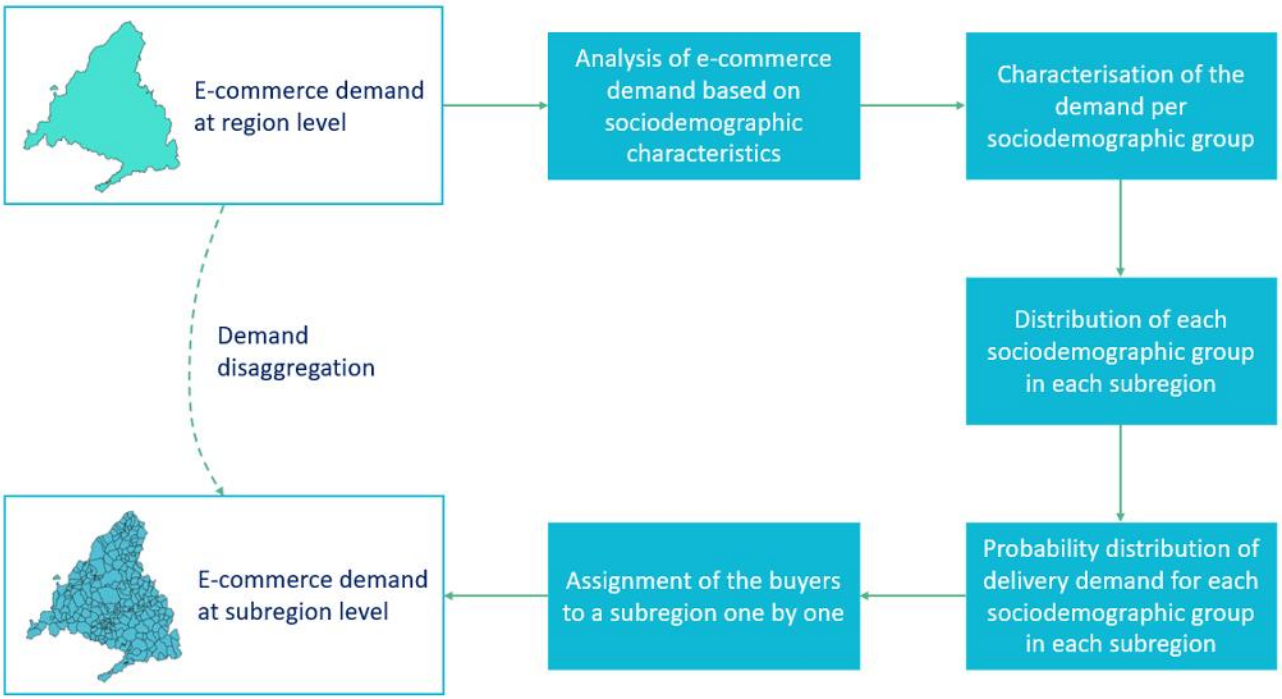

Figure 10 Workflow for the estimation of delivery demand.

4.2.1.4 Methodology for Phase 2

The objective of phase 2 is to deepen the characterization of the parcel delivery demand of a zone, providing not only its demand volume, but also identifying delivery trips and flows (including typical travel times and distances), characterising last-mile delivery-related traffic.

Figure 11 shows the methodology defined for the identification of delivery trips from MND. This methodology is based on a pattern transfer from the delivery trips data to the MND data. For that, the e-commerce delivery data is used to extract mobility patterns based on trip features observed in delivery trips:

- average travel distance in the day: this variable is expected to be high for 4 to 5 days a week and normal for the other days (in which the professional is resting),
- average travel distance between deliveries: this information allows the distinction between stops (deliveries) and breaks,
- radius of gyration from the logistic centre: this variable is expected to be smaller than a fraction of the diameter of the region analysed, as we are considering last-mile delivery (i.e., short-distance transport),
- average number of deliveries,
- average travel time in the day: this variable is expected to be high for 4 to 5 days a week and normal for the other days (in which the professional is resting),

- average travel time between deliveries: this information allows the distinction between stops (deliveries) and breaks,
- frequency of appearance in logistic centres,
- average working hours: this variable is expected to be between 8 and 10 for 4 to 5 days a week.

These patterns are crossed with the activity-travel diaries obtained from MND to identify those users whose activity-travel diaries match the obtained patterns.

This methodology is calibrated and validated using the delivery demand data for Madrid provided by Citylogin (see Section 4.2.1.2).

Once this development is finished, the capability of the historical delivery trips data for demand prediction will be also analysed. For that, historical delivery trips will be identified using this approach, and the delivery demand of the next days will be predicted based on the demand of the previous days.

Following the from-particular-to-general approach, this methodology can be applied to any case in which it is required to identify delivery trips from geolocated data (such as MND or GPS data), provided that a sample of delivery trips data is available to extract representative mobility patterns of professional drivers of the region.

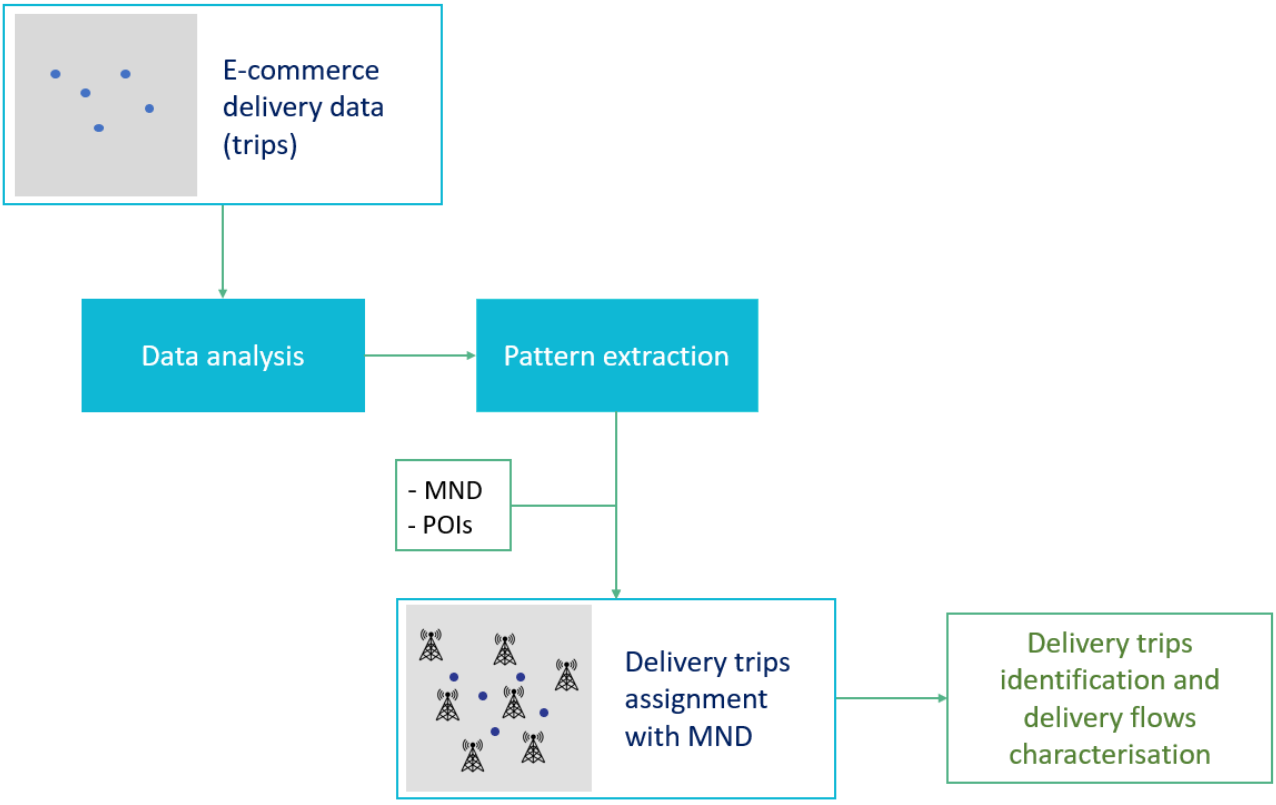

Figure 11 Workflow for the identification of delivery trips.

4.2.1.5 Technical Implementation

At this stage of the project, only phase 1 has been implemented. For that, the Survey on Equipment and Use of Information and Communication Technologies in Households of 2023 provided by the INE is being used. This survey contains information about the use of e-commerce for private

reasons. In particular, it provides the number of times a user has made an online purchase in the last 3 months, and the kind of products bought, distinguishing between physical products (clothes, shoes, jewellery, toys, sports articles, music, books or films in physical format, computers, mobile phones, food, furniture, etc.) and non-physical products (tickets, online videogames, apps, etc.). This distinction is quite useful, as only e-commerce of physical products is needed to analyse the delivery demand.

The questions of the survey refer to the last 3 months (from the moment it is answered), and it was carried out between May and August 2023, hence, the answers refer to the period from February to August 2023, all of them regular months without any relevant festivity in Spain (such as Christmas). So, it can be assumed that the online shops correspond to three regular months of the year, and by dividing them by 3, we obtain the online shops for a regular month of the year.

The respondents are characterised by their age, gender, nationality, province of residence, level of studies, household structure, household net income, employment situation, and marital status.

As mentioned in Section 4.2.1.3, the methodology for phase 1 is being validated for the Madrid region, with a population of 6,750,336 inhabitants in 2022 (the latest available data). In this case, the demand for the region was disaggregated for each of the 246 districts.

According to the survey, 3,205,371 persons used e-commerce to buy physical products at least once during three regular months (almost half of the inhabitants), with a total of 14,846,565 purchases.

The first step for the demand disaggregation is the definition of the sociodemographic groups to be used. For this first iteration of the implementation only age and gender were considered for disaggregation, as only average values of net income and household sizes information are available at subregion level. Hence the sociodemographic groups are defined as all the possible combinations of the following categories of the available age and gender characteristics:

- Gender: 2 groups: male/female.
- Age: 9 groups: 15-24, 25-34, 35-44, 45-54, 55-64, 65-74, 75-84, 85-100.

The gender categories are the same as they appear in the Census, while for the age categories an aggregation has been made to reduce the number of groups, hence easing the results interpretation. When performing this aggregation it is important to ensure that they effectively characterise population groups with different behavioural patterns, i.e., people belonging to the same group effectively behave in a similar way, and people of different groups have different online shopping patterns. Moreover, the groups must be commensurable with the minimum aggregation provided in the census data, which have a granularity of 5 years. The group for 0-14 years old is removed since in the survey participants age goes between 16 and 100 years (people younger than 16 years old are not allowed to buy online, as they are not allowed to have a credit/debit card).

To compute the distribution of each sociodemographic group, we use the census data provided by the INE for the year 2022, as the one for 2023 is not available until the year is finished. However, this is not a problem since census information does not change significantly between two regular consecutive years. Next, the census data are aggregated to extract the distribution of the different groups in Madrid. This is depicted in Figure 12.

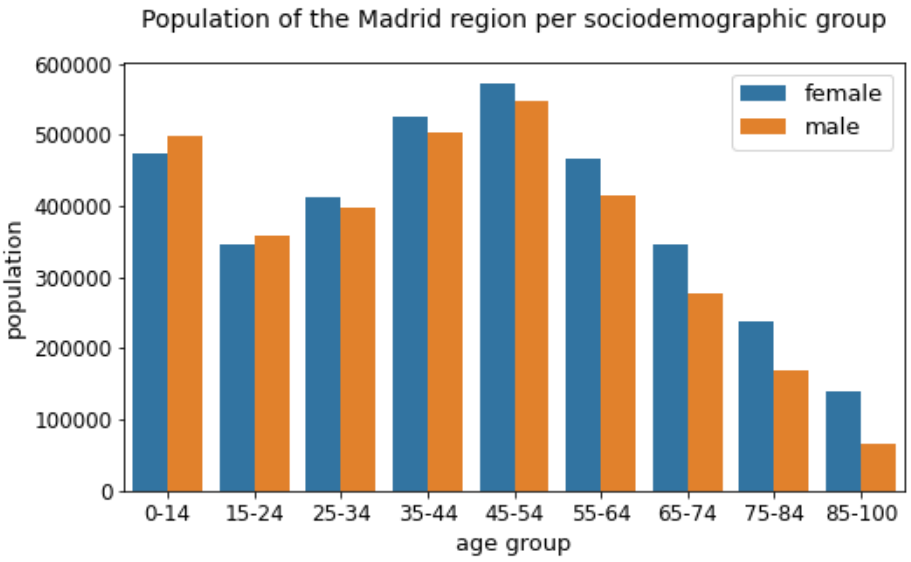

Figure 12. Population of the Madrid region per sociodemographic group.

Then, the distribution of buyers and deliveries per sociodemographic group is extracted from the survey. Figure 13 shows this distribution in the Madrid region, according to the survey of 2023. The first row depicts the distribution of the total number of deliveries and buyers per sociodemographic group, and the second row shows the ratio of deliveries and buyers with respect to the total population of each group. As can be seen, the absolute values and the ratios follow the same distribution except for the age group between 25 and 35 years old, whose members are the ones that make the most use of e-commerce, in proportion. Also, up to 54 years old women are more likely to shop online for physical products than men, while from 55 onwards, the trend is reversed.

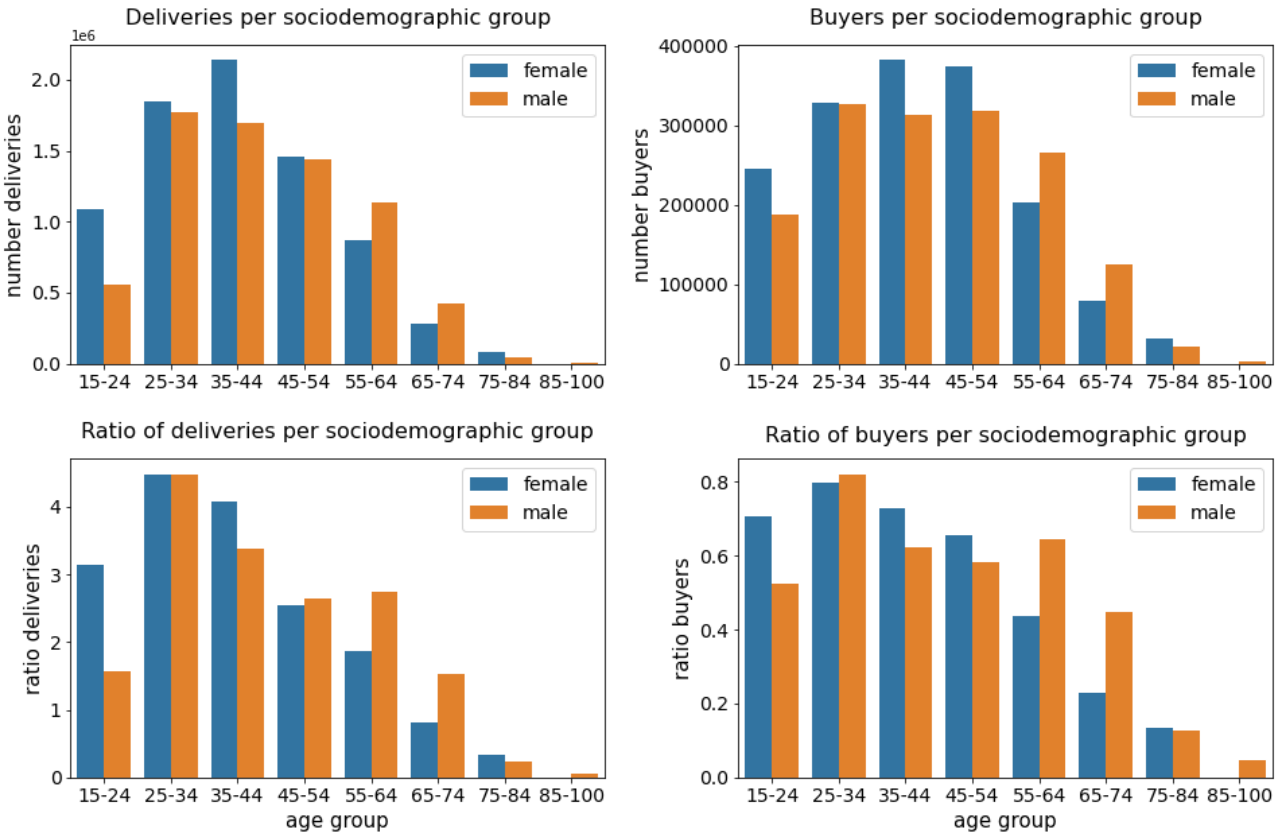

Figure 13 Distribution of deliveries (left column) and buyers (right column) per sociodemographic group. The first row shows absolute values and the second row, the ratio with respect of the total population of each group.

As can be seen, the population and the buyers follow a similar distribution per sociodemographic group, as expected. Then, the population per district and group is divided by the total population of the group in the region. This proportion distribution is used as seed probability distribution to dynamically assign the buyers one by one to a district, according to their sociodemographic group.

As an example, Figure 14 depicts the initial probability distribution among districts of women and men of age groups 15-24 and 65-74. As can be seen, the central area accounts for most of the population.

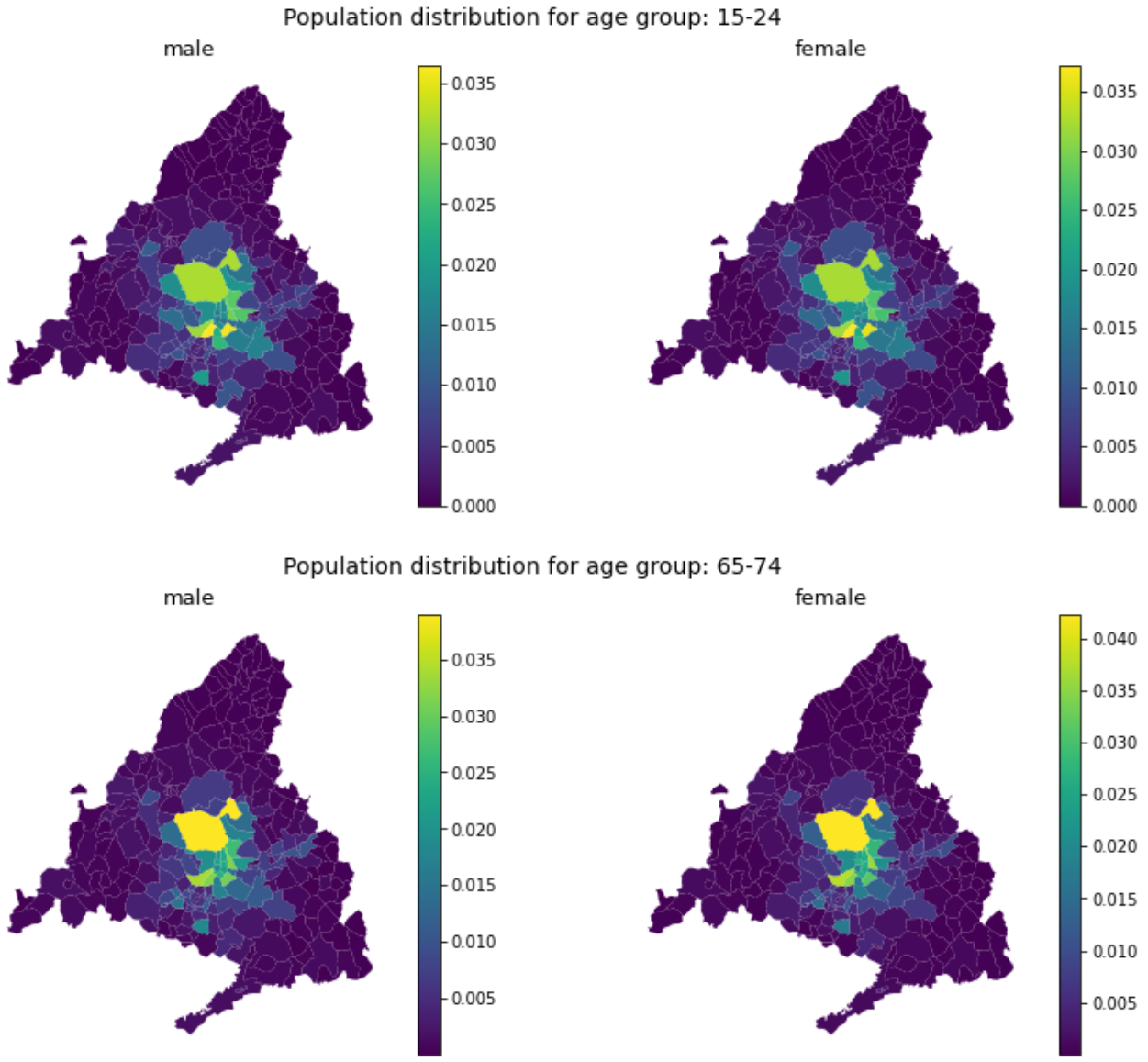

Figure 14 Population distribution in the Madrid region of women and men of age groups 15-24 and 65-74.

The next section shows the results of the demand assignment to district.

4.2.1.6 Results

Figure 15 and Figure 16 show the distribution of buyers and deliveries among the districts of the Madrid region per sociodemographic group. As can be seen, both distributions are similar, as there is a strong relation between the number of buyers and the number of deliveries.

As expected, the results show that the population group and population density highly determine the number of orders a person makes.

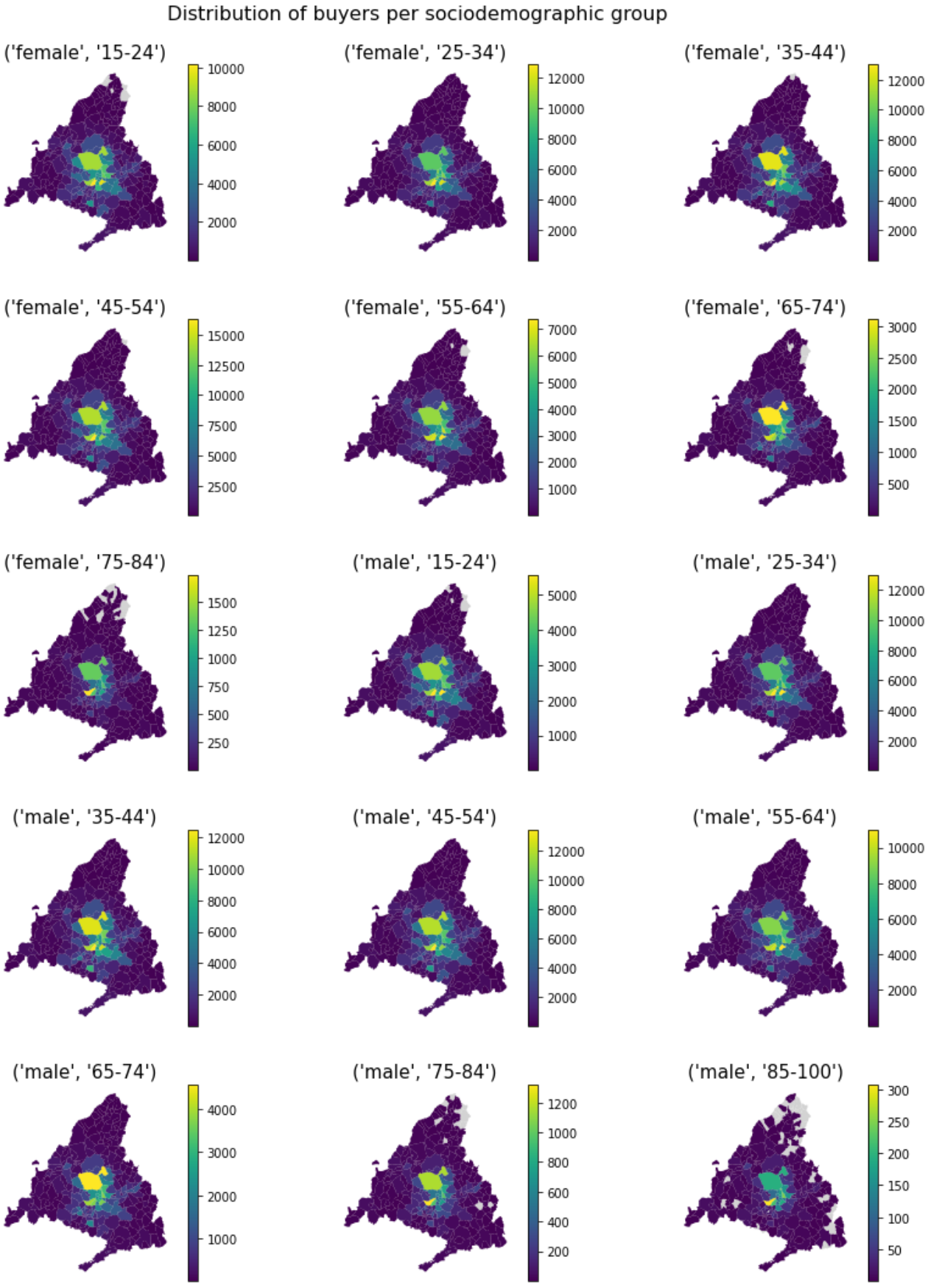

Figure 15 Distribution of buyers per sociodemographic group.

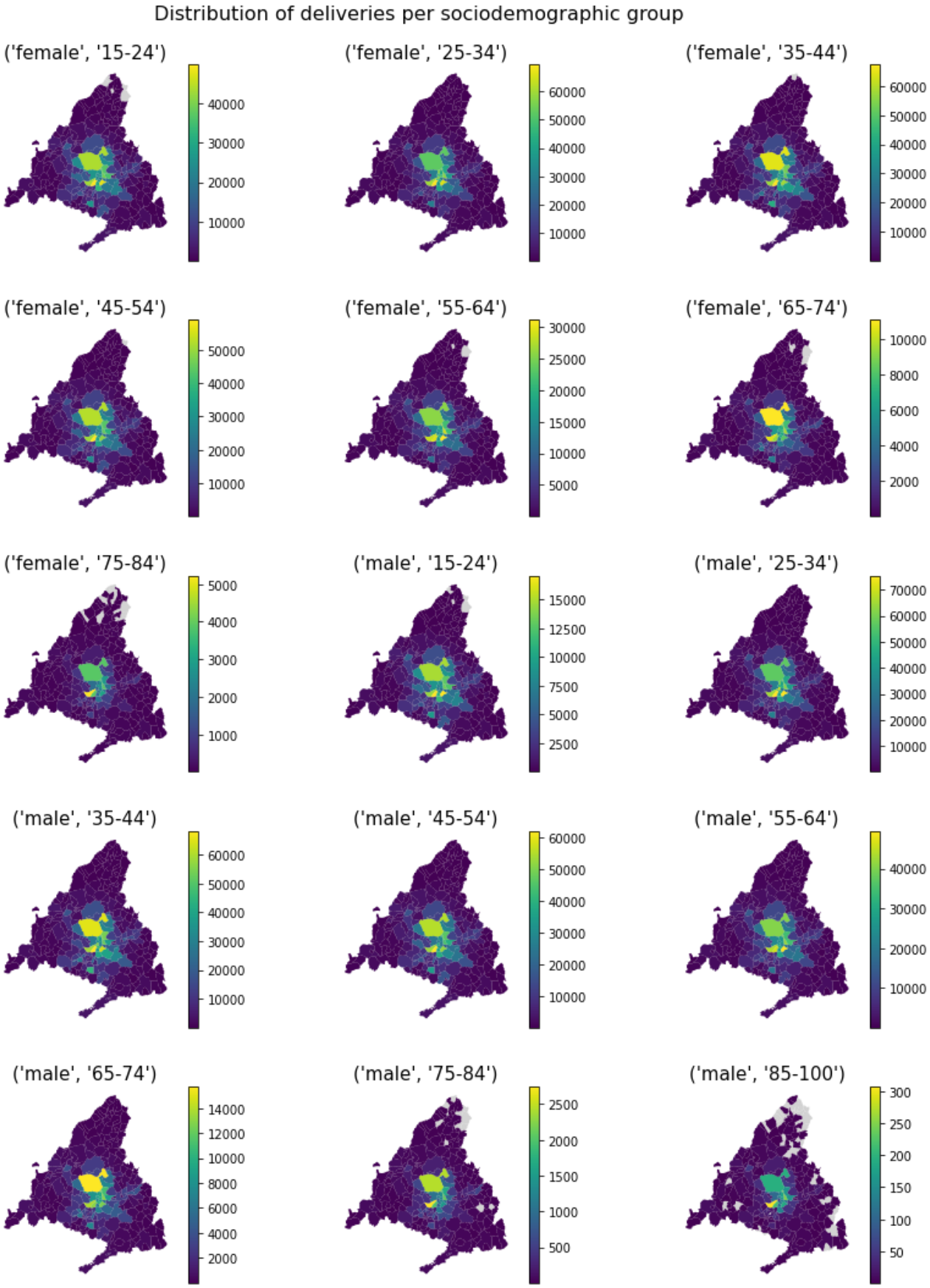

Figure 16 Distribution of deliveries per sociodemographic group.

4.2.1.7 Next steps

As next steps of the implementation of phase 1, the physical products considered in the survey will be grouped to separate the ones compatible with passenger transport from the ones not compatible. The demand of the compatible ones will be disaggregated and analysed, as this is the one needed for CONDUCTOR UC3.

Also, it will be studied how other characteristics can be included in the disaggregation, especially household net income and household size. For that, additional data would be needed.

Finally, the deliveries for an average month of previous years will be computed (the e-commerce INE survey is available from 2003) and considered as time series to analyse the evolution of the delivery demand and the effect of the COVID-19 crisis in the e-commerce. The capability of the historical data for demand prediction of a regular month of the next year will be also analysed.

4.2.2 Identification of Unusual Traffic Patterns Caused by Large-scale Events

4.2.2.1 Introduction

In recent years, event detection using public data has received a great deal of attention from researchers and practitioners. In particular, the widespread availability of social media data (e.g., Twitter data) resulted in a significant increase in studies that are using machine learning algorithms and natural language processing techniques to extract knowledge from micro texts in social networks and identify events of interest (e.g., Belcastro et al., 2021; Ferreira da Silva et al., 2022; Ganeshkumar et al., 2022). In addition to this, a special emphasis started to be placed on geotagged social media, where the main goal is to mine social media data streams and recognize events in a specific local or global area of interest (e.g., Afyouni et al., 2022; Hodorog et al., 2022; Vitanza et al., 2023; Afyouni et al., 2023).

In the context of the CONDUCTOR project, INTRA has proposed a traffic event detection mechanism based on machine learning and geo-visualization to identify traffic events and trace the development of these events in real-time. Specifically, taking advantage of sensors and social networking platforms, the traffic event detection tool aims to provide first responders with the right information to create situational awareness. This information can be a list of metrics helping first responders to identify extreme events (e.g., traffic accidents, natural disasters, social events, unusual happenings, etc.) in the road network and monitor traffic conditions. Providing first responders with the right information at the right time will help them to take appropriate actions reasonably and facilitate decision-making under risk and uncertainty. Therefore, the aforementioned metrics can be seen as early warning signals for monitoring and improving traffic conditions in road networks.

4.2.2.2 Methodology & Technical Implementation

Despite the advantages of social networking platforms, social media data heterogeneity and big data size pose challenges in the process of identifying information about events from the raw data. Most of this data is unstructured and includes text in different formats. How to capture reliable, valuable and accurate information in massive data is one of the most significant research topics nowadays (Rashinkar & Krushnasamv, 2017). Specifically, big data is accompanied by difficulties and challenges (e.g., data imperfection, data inconsistency, data confliction, data alignment/correlation, etc.) in a data-driven service provision due to its “5Vs”, namely volume, velocity, variety, veracity and value (Meng et al., 2020). Considering the data heterogeneity obtained on sensors and social networking platforms, we propose a four-level fusion pipeline combining data fusion concepts (Mitchell, 2012) with machine learning (Khaleghi et al., 2013) to address data fusion challenges effectively concerning criteria such as efficiency, quality, stability, robustness and extensibility. The

suggested fusion pipeline depicts the main aspects of the methodology we are going to follow for designing and implementing the traffic event detection mechanism mentioned in Section 4.2.2.1.

The four-level fusion pipeline consists of four main modules (one at each level): data acquisition for storing and data retrieval purposes, data fusion for making data more accurate, information fusion for extracting significant features/predictors and decision/knowledge fusion for supporting decisions (see Figure 17).

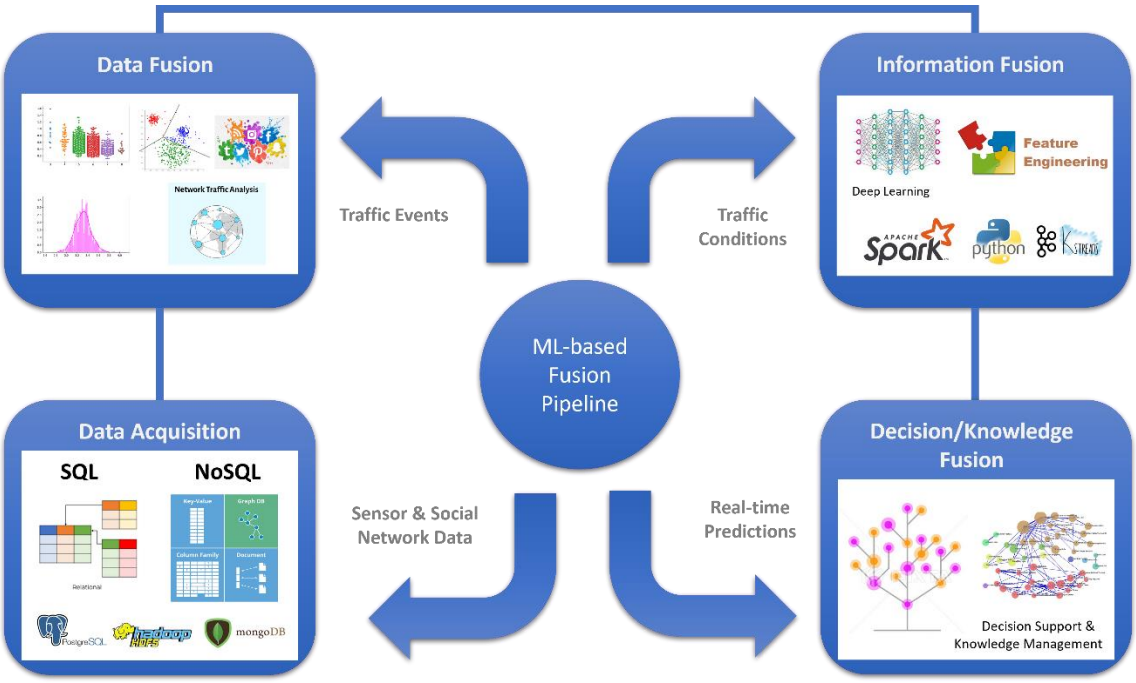

Figure 17 Three-level fusion pipeline.

To begin with, SQL (e.g., PostgreSQL) and NoSQL (e.g., MongoDB) databases can be available for storing and managing structured transactional and relational datasets, as well as unstructured data, respectively. At this stage, we mostly focus on gathering data based on sensors used on the CONDUCTOR project, as well as social networking platforms that allow data retrieval through APIs. The first objective of the data fusion module is to create a pool of techniques for addressing big data challenges effectively and ensuring a data value chain that allows someone to produce a cleaned, unbiased and accurate dataset for model development purposes. In this direction, data are explored in depth by providing an analyst with visualization techniques and data quality measures that can help him/her identify any issues or bad data. The second objective of the data fusion module is to provide efficient computational intelligence approaches to identify traffic-related data and extract traffic events. Natural language processing practices, clustering techniques, as well as anomaly detection algorithms will be investigated to pre-process structured/unstructured data, investigate similarities and identify outliers/abnormal behaviours in the traffic-related data.

The information fusion module contains advanced statistical learning approaches for extracting new features based on available variables, allowing us to further explore datasets and find hidden patterns. In this direction, feature weighting techniques will be investigated for training machine learning algorithms with high predictive accuracy. This module will be very useful for developing additional models that can support the traffic event detection mechanism. For instance, demand forecasting in specific paths of a road network could be additional data to be considered by the traffic event detection mechanism.

The decision/knowledge fusion module supports the decision-making process by providing someone with trained ML models for detecting and predicting traffic events, as well as monitoring traffic

conditions. In the context of this module, a special emphasis is put on trustworthy and explainable ML, explaining ML outcomes to users in a language close to a human expert. The role of explanations in data quality and ML outcomes is very crucial especially when it is needed to address the issues of transparency (Gunning et al., 2019; Bertossi & Geerts, 2020). As a result, with this module, we support a hybrid-augmented human-in-the-loop process where computational intelligence (data science practices) and human intelligence (expert knowledge) are combined under seven aspects: (a) the ability to perceive rich and complex information from traffic-related data, (b) the ability to discover causal information from observational data, (c) the ability to learn in a particular context of interest, (d) the ability to abstract, (e) the ability to create new meanings/concepts, (f) the ability to reason for decision-making, and (g) the ability to explain the prediction/decision outcome. In this direction, insightful graphs and visualizations will be suggested.

4.2.2.3 Results

This section aims at presenting the main models developed for supporting data/information/knowledge fusion concepts. To begin with, data fusion module includes an automatic anomaly detection algorithm based on tree-based approach, namely Isolation Forest (Liu et al., 2008). In particular, the Isolation Forest model randomly selects a feature (variable) from the dataset and then randomly selects a split value between the maximum and minimum values of the feature (variable). In this direction, it is feasible to isolate and calculate the isolation path for every sample in the dataset. The Isolation Forest computes two metrics, a binary anomaly indicator where 1 means that an anomaly/outlier identified and 0 otherwise, as well as an anomaly score belonging to the interval [-1,1]. A value from 0 to 1 indicates an anomaly, whereas a value less than 0 and near to -1 indicates that no anomaly/outlier exists.

As an illustrative example, we used road traffic data of the region of Attica, Greece, that are available at https://data.gov.gr/datasets/road_traffic_attica/. The dataset consists of the following variables:

- *deviceid*: the id of the sensor.
- *countedcars*: the number of cars counted.
- *appprocesstime*: a timestamp.
- *road_name*: the name of the road.
- *road_info*: more details for the road.
- *average_speed*: an average speed detected.

In the context of the information fusion level some feature engineering tasks took place. Specifically, according to the timestamp, the name of the day within a week can be detected, whereas according to the time, different time zones can be created (e.g., early morning, morning, early afternoon, etc.). Moreover, as the combination of “deviceid”, “road_name” and “road_info”, is unique, we keep only the “deviceid” variable in the dataset. After applying a feature engineering (i.e., create new variables considering existing ones), the new dataset consists of the following features (variables):

- *deviceid*: the id of the sensor.
- *countedcars*: the number of cars counted.
- *average_speed*: an average speed detected.
- *day_name*: the name of the day within a week.
- *time_zone*: the time zone within a day (i.e., early morning, morning, early afternoon, afternoon, early night, night)

The dataset consists of 66999 rows, of which 53599 rows used for training the Isolation Forest model, whereas 13400 rows used for test purposes. It is worth mentioning that all categorical variables were quantified using a Label Encoder.

Considering training dataset, the following figures (Figure 18) depict:

I. A 3D graph (on the left) using the t-distributed stochastic neighbour embedding method (Hinton & Roweis, 2002; van der Maaten & Hinton, 2008) for visualizing the four-dimensional data by giving each data point a location in a three-dimensional map. The method models each four-dimensional object by a three-dimensional point in such a way that similar objects are modelled by nearby points and dissimilar objects by distant points with high probability.

II. A 2D graph (on the right) using the Uniform Manifold Approximation and Projection (UMAP) dimension reduction technique (McInnes & Healy, 2018) that can be used for visualization similarly to t-SNE, but also for general non-linear dimension reduction.

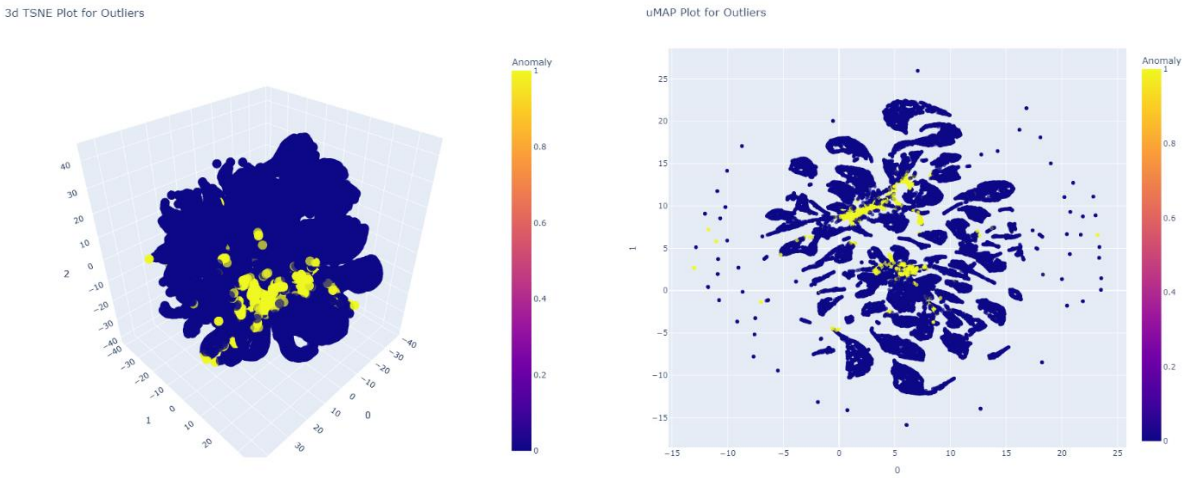

Figure 18 3D t-SNE and 2D UMAP plots.

Moreover, focusing on computed anomaly scores, the following figures (Figure 19) depict the distribution of anomaly scores for training and test datasets, respectively.

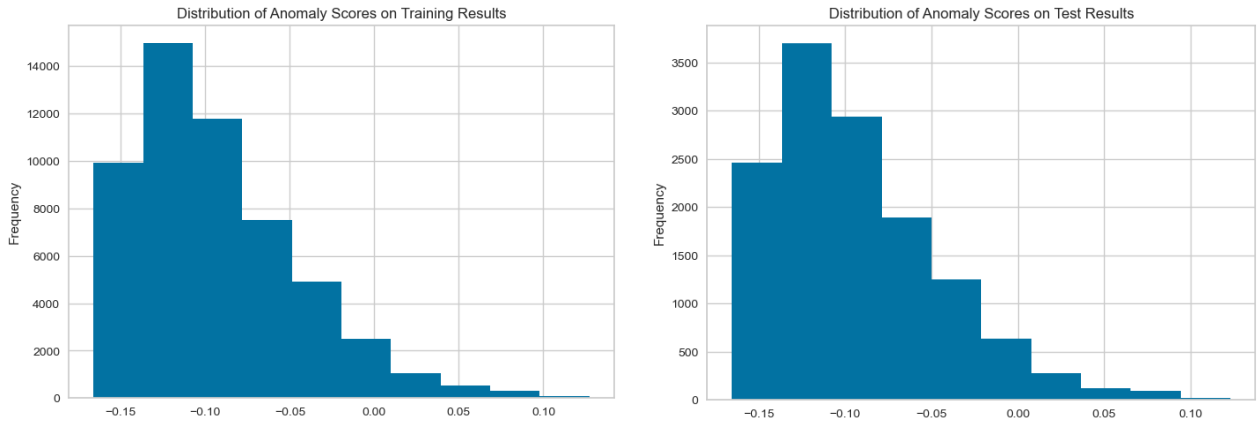

Figure 19 Distribution of anomaly scores for training (left) and test (right) datasets.

It is worth mentioning that an emphasis should be placed on the anomalies/outliers detected. Therefore, focusing on outliers, several patterns can be identified. For example, the following figures (Figure 20) depict the distribution of anomaly scores for the outliers identified in the test set, as well as a line plot comparing number of cars and average speed focusing on a specific day within a week for a month period. Observing the second graph, an emphasis should be put on cases where the average speed is too low, indicating that the road is overloaded.

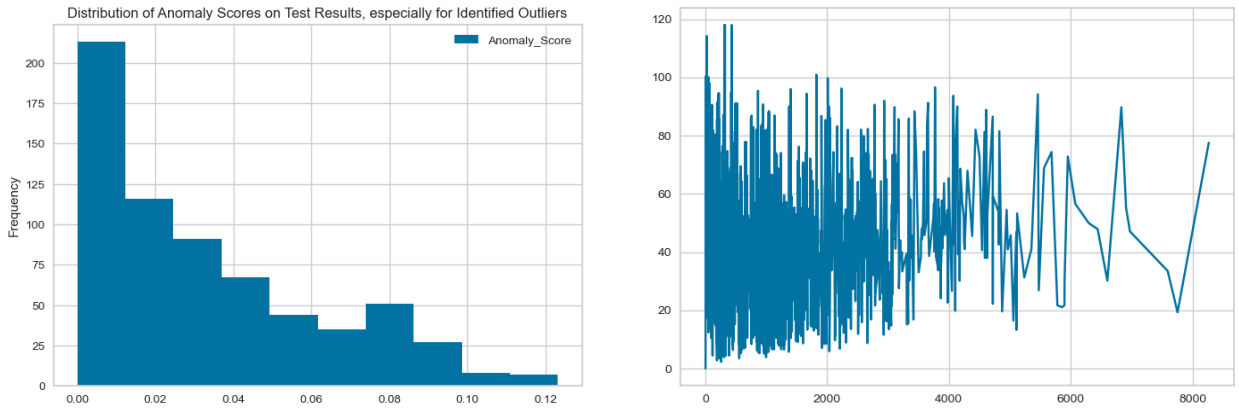

Figure 20 Distribution of anomaly scores of outliers for the test set (left) and line plot (right) comparing number of vehicles and average speed.

A deeper analysis is also feasible. For instance, focusing on specific day of a week (e.g., Friday) and a specific sensor (e.g., MS110), we can visualize the number of cars counted and average speed for each time zone within a day (Figure 21).

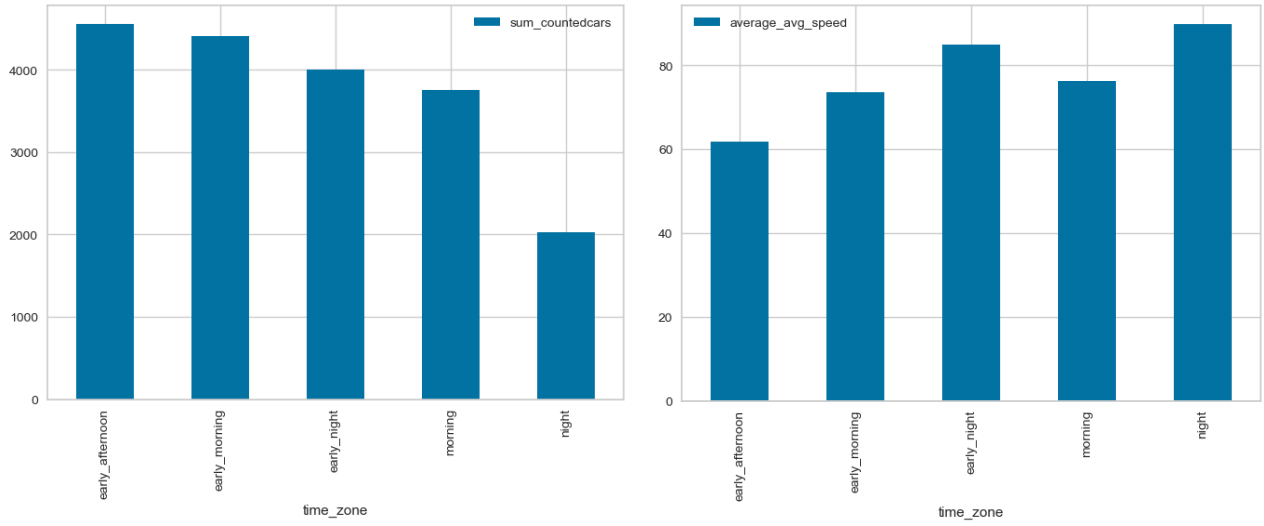

Figure 21 Number of cars per time zone for Friday (left) and average speed per time zone for Friday (right).

As can be observed, in most cases, as the number of cars increased the average speed is decreasing. These cases are of special interest in our example since they depict outliers directly affecting the road network conditions.

Moreover, in the context of the data fusion level, several APIs have been also investigated especially for gathering information regarding scheduled events, e.g., https://www.eventbrite.com/platform/api.

Combining the timestamp of road traffic data and the timestamp of scheduled events we can further extend the dataset by adding a new column regarding the number of scheduled events that will take place in a specific geo-location. In this direction, we introduce a human-in-the-loop hybrid augmented model based on the Mamdani Fuzzy Inference System (MFIS) (Niemiec, 2017; Spolaor et al., 2020) to compute a traffic load risk value belonging to the interval [0, 1]. The MFIS model is part of the knowledge/decision fusion level and aims at combining anomaly scores of outliers and number of scheduled events to calculate an estimation of the risk associated with the traffic load.

MFIS has been developed composing of fuzzy rules with linguistic inputs and outputs so as to obtain rule-based decisions. Inputs reflect main decision variables per event and/or alert (in terms of anomaly score) defined by experts. Values of these variables are continuously monitored for constructing a decision support pipeline that will facilitate experts to make decisions under uncertainty and risk. When the inputs are given, there are six steps to compute the output of the MFIS (see Figure 22).

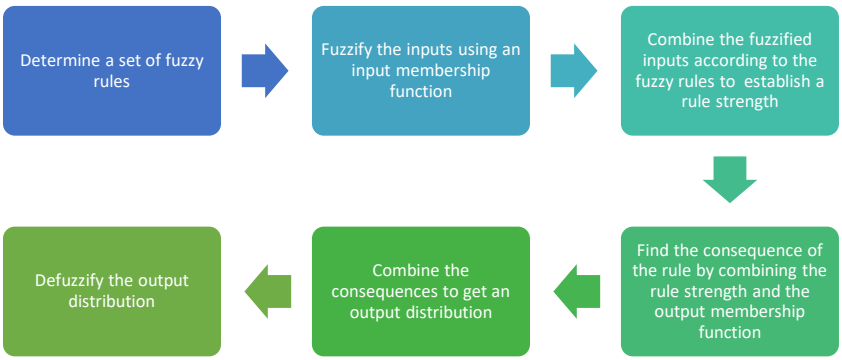

Figure 22 Mamdani Fuzzy Inference Approach.

As far as the inputs are concerned, computed anomaly scores and number of scheduled events identified in the data fusion level are associated with decision variables to be monitored through CONDUCTOR components. These decision variables are used for building rules and measuring traffic load risk. Typically, a fuzzy rule base consists of a set of fuzzy IF-THEN rules depicting the core of the fuzzy inference system in the sense that other components such membership functions are designed to implement these rules in a reasonable, realistic, and efficient manner. These IF-THEN rules are utilized by the fuzzy inference system to determine a mapping from fuzzy sets in the input universe of discourse $U \subset R^n$ to fuzzy sets in the output universe of discourse $V \subset R$, based on fuzzy logic principles. The fuzzy IF-THEN rules are given by the following equation:

$$R^{(v)} = IF\ x_1\ is\ F_1^v\ \dots\ x_n\ is\ F_n^v, \qquad THEN\ y\ is\ G^v\ ,$$

where $F_j^v, G^v, j = 1, \dots, n$ are fuzzy sets in $U_j \subset R$, respectively. In addition to this, $x = [x_1, \dots, x_n]^T \in U$ and $y \in V$ are input and output linguistic variables of the fuzzy inference system which belongs to the input and output universes, respectively. v represents the number of rules in the fuzzy rule base.

Furthermore, fuzzy membership function is used to convert the crisp input provided to the fuzzy inference system. Formally, a membership function for a fuzzy set A on the universe of discourse $x \in U$ is defined as $\mu_A: x \to [0, 1]$, where each element of x is mapped to a value between 0 and 1. This value is called membership value or degree of membership, quantifying thus the grade of membership of the variable $x_j \in x, j = 1, \dots, n$ to the fuzzy set A. Namely, x is the universal set, whereas A is the fuzzy set derived from x.

In the context of the proposed fuzzy-based solution approach, the triangular membership function is used to model anomaly scores and number of schedules events, respectively. The triangular

membership function which fuzzifies the input can be defined by three parameters: a, b and c where a and c defines the base, whereas b defines the height of the triangle (see Figure 23).

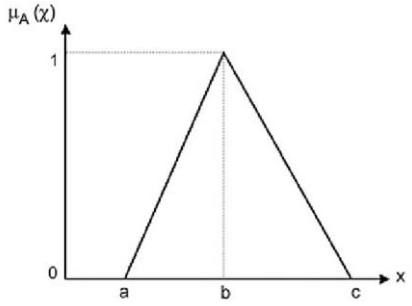

Figure 23 Triangular Membership Function.

X-axis represents the input from the process, whereas y-axis represents corresponding fuzzy value. Analytically, if $x_j = b$, then it is having full membership in the given set, that is $\mu(x_j) = 1, if\ x_j = b, j = 1, \dots, n$. Additionally, if input is less than a or greater than c, then it does not belong to fuzzy set at all, and its membership value will be 0, that is $\mu(x_j) = 0, if\ x_j < a\ or\ x_j > c, j = 1, \dots, n$. If now $x_j, j = 1, \dots, n$ is between a and b, its membership value varies from 0 to 1. If it is near to a, its membership value is close to 0, and if its membership value is near to b, its membership value gets close to 1: $\mu(x_j) = \frac{x_j - a}{b - a}, a \le x_j \le b$. Finally, if $x_j, j = 1, \dots, n$ is between b and c, its membership value varies from 0 to 1. Specifically, if variable is near to b, its membership value is close to 1, and if it is near to c, its membership value gets close to 0: $\mu(x_j) = \frac{c - x_j}{c - b}, b \le x_j \le c$. Mathematically, the triangular membership function is formulated as:

$$\mu(x_j; a, b, c) = \begin{cases} 0, & x_j \le a \\ \dfrac{x_j - a}{b - a}, & a \le x_j \le b \\ \dfrac{c - x_j}{c - b}, & b \le x_j \le c \\ 0, & x_j \ge c \end{cases} = \max\left(min\left(\frac{x_j - a}{b - a}, \frac{c - x_j}{c - b}, 0\right)\right)$$

where a, b and c are defined by experts.

In addition to this, let μ_A and μ_B be membership functions that define the fuzzy sets A and B, respectively on the universe X. To evaluate the disjunction of the rule inputs, an OR fuzzy operator (representing the union of fuzzy sets) is defined as follows:

$$\mu_{A \cup B}(X) = \max\left(\mu_A(X), \mu_B(X)\right)$$

Similarly, to evaluate the conjunction of the rule inputs, an AND fuzzy operator (representing the intersection of fuzzy sets) is defined as follows:

$$\mu_{A \cap B}(X) = \min\left(\mu_A(X), \mu_B(X)\right)$$

Finally, it is worth mentioning that the complement of a fuzzy set A is fuzzy set defined by the membership function:

$$\mu_{A^c}(X) = 1 - \mu_A(X)$$

The following figure (Figure 24) depicts an illustrative example of the fuzzy rule-based inference approach.

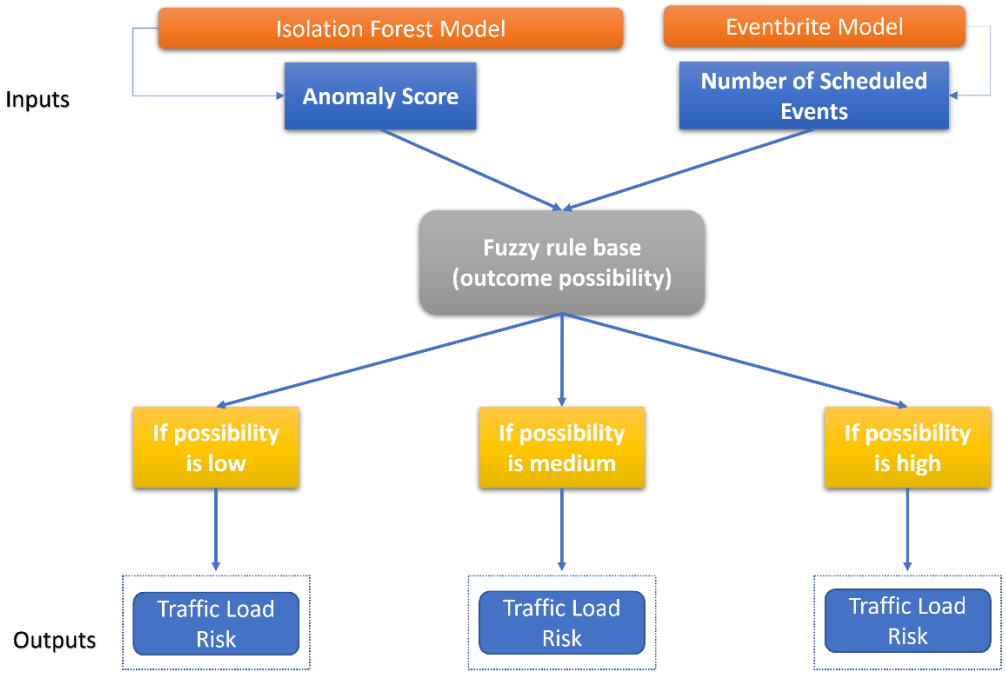

Figure 24 Fuzzy Rule-based Inference Approach.

Inputs of the fuzzy inference approach are two variables: the anomaly score associated with the Isolation Forest Model, and the number of scheduled events related to the Eventbrite API. Each input has a membership function that is defined using expert opinions. For instance, if the value of an anomaly score is between x-value and y-value then anomaly score has a low significance; if the value of anomaly score is greater than y-value then it has a high significance, etc. Considering the result obtained from each variable's membership, an "outcome possibility" can be calculated via the membership function of the output variable (also, based on expert opinions). The "outcome possibility" depicts a set of possible fuzzy rules that are obtained via the combination of the variables' significance to recognize the possible outcome when the membership function of the output variable is applied. An example of fuzzy rule is the following: if anomaly score is low, number of scheduled events is medium, then the "outcome possibility" is medium. Knowing the "outcome possibility", we can provide other partners of the CONDUCTOR with new metrics that can facilitate the decision-making process.

Below, Figure 25 depicts an instance of Triangular Membership Functions related to input variables, whereas Figure 26 depicts an instance of the Triangular Membership Function regarding the output layer, as well as the distribution of traffic load risk on the test dataset.

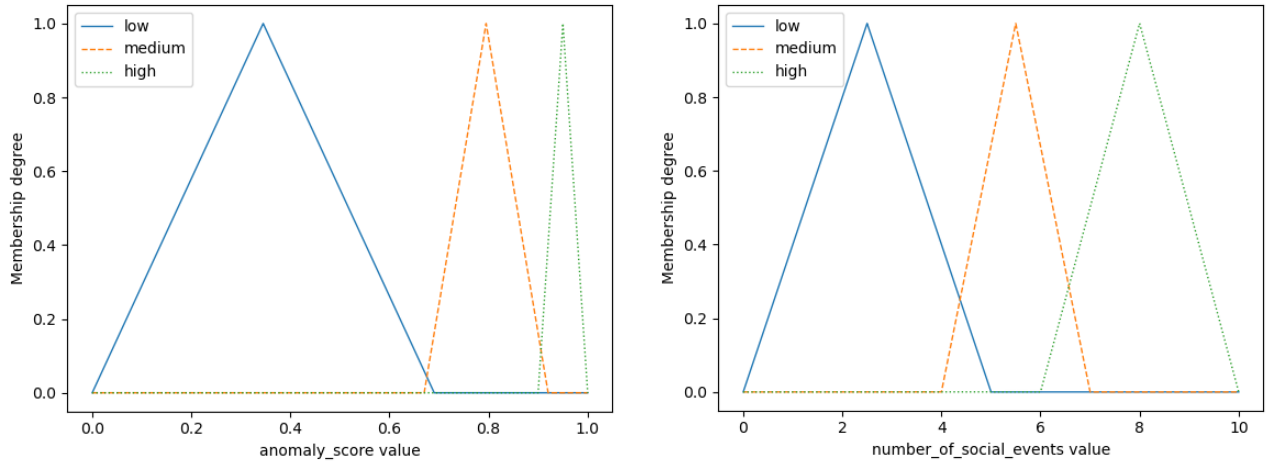

Figure 25 Triangular Membership Functions – Input Layer.

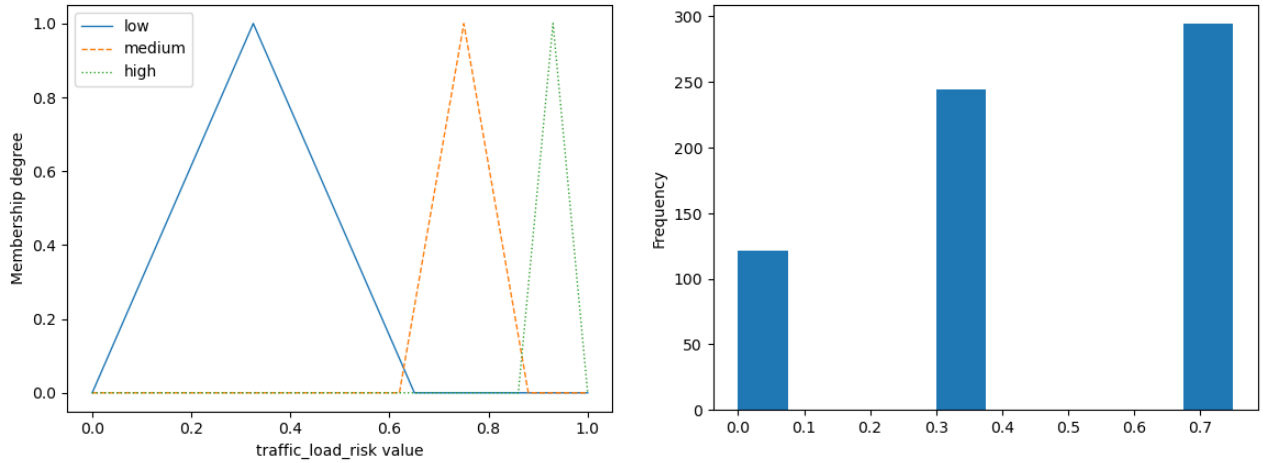

Figure 26 Triangular Membership Function for the Output Layer (left) and distribution of the traffic load risk (right) regarding the outliers/anomalies identified in the context of the test dataset.

To conclude, we suggested the Isolation Forest Model for detecting outliers/anomalies in a road network, whereas taking advantage of public APIs such as the Eventbrite, we enriched the dataset with the number of scheduled events identified in a specific geo-location and time. Both models are the main components of the data fusion level. In addition to this, feature engineering tasks are also available in the context of the information fusion level. Finally, the knowledge/decision fusion level is enriched with a human-in-the-loop hybrid-augmented model based on the MIFS algorithm in which an expert is always part of the system, introducing in such a way the human cognitive capability into computational intelligence algorithms. Therefore, we combine expert intelligence (i.e., knowledge on the domain) with computational intelligence (i.e., computational power) to achieve high accuracy and reliability of proposed solutions. New metrics, such as the traffic load risk, obtained in the knowledge/decision fusion level can be useful for recommendation purposes.

4.2.3 Framework for Actionable Smartphone-based Data Analytics

4.2.3.1 Introduction

The advent of smartphones has revolutionised various aspects of our daily lives, including how we interact with technology and gather information. As versatile technological ecosystems, smartphones are increasingly instrumental in various domains, particularly in transportation analytics (Vlahogianni & Barmpounakis, 2017a). Their multifunctional capabilities, coupled with a broad spectrum of sensors, namely motion sensors (accelerometer, gyroscope, magnetometer), location sensors (GPS, network-based), and ambient sensors (light, microphone, proximity), make them invaluable tools for collecting and analysing transportation-related data.

Having a closer look at the smartphone-based transport literature one can identify three main research domains: driving analytics and recommendations, mobility analytics and parking analytics. The combination of their processing capabilities and embedded sensors like accelerometers, GPS, and cameras, enables the continuous monitoring of driving behaviour and facilitate the identification of extreme driving patterns, such as speeding, harsh braking or acceleration, harsh cornering (left or right turn with high speed), and harsh lane changing (Handel et al., 2014; Johnson & Trivedi, 2011; Mantouka & Vlahogianni, 2022; Predic & Stojanovic, 2015; Tselentis et al., 2017; Wahlström et al., 2015; White et al., 2011).

The principal advantage of utilising smartphones in this context lies in their ability to provide a non-intrusive environment for continuous data collection, offering a more sustainable and cost-effective solution compared to traditional instrumented vehicles. The embedded sensors in smartphones, like GNSS and IMU, are crucial in gathering granular data on driving behaviour, thereby contributing significantly to research in driving patterns and safety (Vlahogianni et al., 2013, 2014). The vast amounts of driving behaviour data collected by smartphones have been systematically used to mine driving patterns under typical conditions as well as major disturbances, such as COVID-19 pandemics (Fafoutellis et al., 2023).

Based on smartphone driving data analytics, many recommendation systems have emerged. A significant branch of this type of research focuses on Advanced Driving Assistance Systems (ADAS) through crowdsourced mobile phone data for improving efficiency and safety (Mantouka & Vlahogianni, 2022) other efforts focus on research on systems promoting, recommending eco-driving and improving driving experience (Campolina et al., 2020; Fafoutellis et al., 2020; Gilman et al., 2015; Magaña & Organero, 2014). Eco-driving recommendations have been shown to improve driving behaviour, encouraging smoother and safer habits (Fafoutellis et al., 2023; Konstantinou et al., 2023). Further, research has centred around scoring methodologies, leaderboards, achievements, and competition that contribute to behaviour assessment and awareness (risk, operational, and economy scores) (Tselentis et al., 2019) and insurance policies based on driving behaviour, including Pay as You Drive and Pay How You Drive systems (Fafoutellis et al., 2022; Tselentis et al., 2017).

Smartphones have also transformed mobility analytics. The evolution of methodological approaches in transport science, from traditional surveys to smartphone-based travel surveys (SBTS) (Servizi et al., 2021; Stopher & Greaves, 2007; Zhao, Pereira, et al., 2015), highlights the growing significance of smartphones in this domain. SBTS offers scalability, supports high-resolution datasets, and allows for a more detailed analysis of transport behaviour over extended periods. Moreover, methodological advancements have enabled the detection of frequently visited locations, identification of primary and secondary activities, and the construction of daily trip chains with capability for quantifying in detail teleworking and home-based activities and parking visits (Jay et al., 2022; Mourtakos et al., 2023).

In the realm of parking analytics, smartphones are proving to be invaluable in reducing urban road congestion. By modelling the search duration for parking spaces using smartphone data, researchers have been able to identify key factors influencing parking search times (Krieg et al., 2018; Mantouka et al., 2021; Salpietro et al., 2015). Studies have employed advanced techniques like parametric and semi-parametric survival models, random survival forests, and deep learning models to predict parking search duration estimation, using data enriched with variables like population density and land use. The detection of cruising (searching for parking) using GPS data from smartphones has also been a significant advancement. Studies have proposed new methods for detecting cruising and used machine learning algorithms to forecast cruising times in different urban areas, demonstrating the potential of smartphones in providing real-time, data-driven solutions to urban parking challenges.

The pathway from designing naturalistic experiments using smartphones to evidence-based decision making based on crowdsourced information from smartphones is paved with a variety of challenges summarised in Table 2.

Table 2 Existing challenges in analytics using smartphones and suggested countermeasures

Challenge	Suggested countermeasures	Selected Citations
Enhancing data representativeness through user engagement	·Incentives and gamification aspects ·Convince the crowd for the usefulness and importance of driving behaviour understanding in traffic and road safety ·Advanced annotation tools to facilitate engagement and reporting ·Reducing cost per user for data collection by leveraging novel lowcost technologies	(Nitsche et al., 2014; Stopher & Greaves, 2007; Vlahogianni & Barmpounakis, 2017b; Yen et al., 2019)
Ensuring for data availability and quality	·Utilize low-power wireless networks ·Upload data when a Wi-Fi connection is available ·Share sensing data among multiple systems ·Change sampling rates ·Anonymity, pseudonymity, spatial cloaking ·Data perturbation and aggregation ·Feature selection (Filter, wrapped methods) ·Anomaly detection techniques ·Assess the necessary amount of data for capturing user behaviour	(Christin, 2016; Etemad et al., 2018; J. Wang et al., 2018; Stavrakaki et al., 2020; Thomas et al., 2018)
Identify the context from the data	·Data fusion ·Filtering algorithms ·Feature Engineering ·Mode detection techniques ·Trip chain detection	(L. Wang et al., 2019; Thomas et al., 2018; Wahlström et al., 2015; Zhao, Ghorpade, et al., 2015)
Detect abnormal patterns	·Machine learning approaches ·Determine universal thresholds for each feature	(Bejani & Ghatee, 2018)
Modelling efficiency, transfer learning and explainability	·Apply resampling techniques ·Generate synthetic samples ·Transfer learning ·Additional features ·Big data analysis instead of small experimental datasets ·Outlier detection ·xAI	(Fafoutellis et al., 2022; Hu et al., 2018; Konstantinou et al., 2023; Maldonado & López, 2018; Roy et al., 2018)
Raising awareness and changing attitudes	·Incorporate ADAS schemes ·Gamification ·Impact assessment studies	(Adamidis et al., 2020; Mantouka et al., 2021; Tselentis et al., 2017; Vlahogianni & Barmpounakis, 2017a)

Real-time operation	·Artificial Intelligence and edge computing ·Prioritize tasks ·Efficient memory management	(Shukla et al., 2018)

NTUA has established a generic detailed modelling plan to address issues of data processing and analysis for stream data coming from smartphone sensors, that creates actionable information out of raw data, contributing to various fields including driving analytics, mobility analytics, and parking analytics. This methodology will be applied in UC1-Athens.

The framework proposed is described in the next section.

4.2.3.2 Framework

Establishing a detailed modelling plan to address issues of data processing and analysis for stream data coming from smartphone sensors involves several key stages with specific focus areas and characteristics (Laña et al., 2021). These stages pertain to sensing, pre-processing and modelling as well as model exploitation and adaptation. Along this pathway (Figure 27 raw data are transformed into actionable information, providing insights into the future, and allowing traffic engineers, organisations and businesses to extract value out of them.

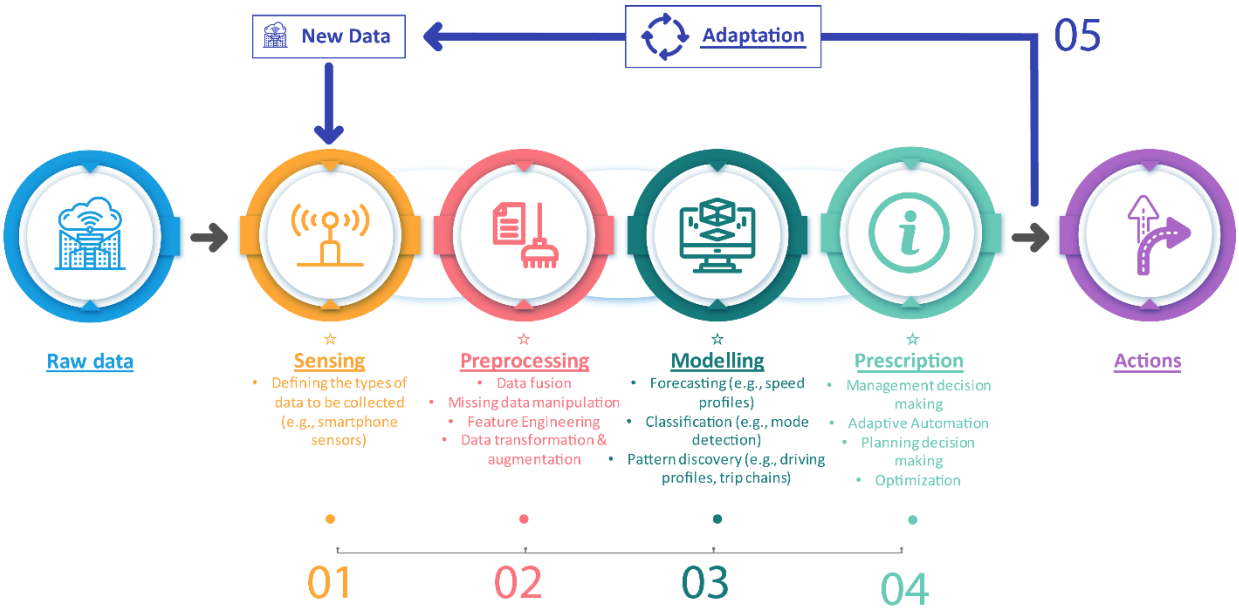

Figure 27 Modelling plan for transforming raw data to actionable information.

1.Sensing

The initial step of the modelling plan is about capturing a wide range of information from smartphone sensors and other sources. This includes traditional sources like vehicle and traffic monitoring and emerging sources such as social media. The primary challenges in this stage include managing the variety, volume and quality of data and selecting relevant data tailored to specific applications. The data sources are categorised as follows:

- Roadside Sensing: Data on speed, traffic flow, and vehicle detection directly from the road.
- In-vehicle Sensing: Data gathered from vehicles; used for fleet management, route optimization and behaviour analysis.
- Cooperative Sensing: Crowdsourced information and social media.

- External Data Sources: Weather conditions, events and socio-economic indicators.
- Structured/Static Data: Data from controlled sources like public transportation schedules and municipalities.

These diverse data types are crucial for ITS applications expanding from traffic pattern analysis to the development of autonomous vehicles, however, the suitability of each data source varies depending on the specific application and effective development greatly depends on this early stage of careful and relevant data selection.

2. Data Preprocessing

The diversity of sensing sources might offer opportunities, but it also brings more than a handful of challenges to the table. The variations in the data regarding form, format, timing and accumulation rate, request some advanced preprocessing skills and tools before modelling can take place. While overlooked in numerous occasions, the preprocessing step is necessary for addressing issues like missing and/or corrupted data which can significantly skew model outcomes. For countering these problems, researchers employ various strategies for data imputation and correction ranging from just cleaning the data, to enriching the data for improved modelling, including data transformations for consistency and relevance, and selecting or engineering features that best fit the modelling needs. In imbalanced datasets, handling these class imbalances is also added in the mix.

One preprocessing's main aspect is data fusion, which despite its potential to create models with higher accuracy and explainability by combining multiple sources remains underexplored. However, it is the limited range of sources that ITS systems are relied on, that makes the exploration of data fusion approaches crucial for the actionability and effectiveness of the model itself.

3. Data Modelling

After collecting and preparing the data the next phase is to start building models that can extract insights through analysing. The purpose of these models may expand from clustering unsupervised data for enhanced value using classification or regression for pattern recognition in supervised data to forecasting future trends based on past data and simulating outputs for better understanding of input data processes. The model choice depends on objectives and combinations of various machine learning methods is common. The key is ensuring that models can generalise well to new, unseen data, balancing between accurate performance on current data and adaptability to new situations.

The complexity met in traffic and transportation operations is usually treated with heterogeneous modelling approaches that aim to complement each other to improve accuracy. These may pertain to trying out many models and selecting the most appropriate one, or combining multiple models at the same time in order for a single output to be given. Incorporation of physical models, like those based on traffic theory, into data-driven models can also enhance accuracy and applicability. The process often involves optimization of model hyperparameters, sometimes using advanced techniques like Evolutionary Computation or Swarm Intelligence. However, as the complexity of the models increases, so does the challenge of optimising these parameters. It is important to note that with more complex models, achieving a completely deterministic and stable solution is rarely possible.

4. Model Exploitation (Prescription)

After modelling phase, the next step is the application of the developed model to real-world scenarios. This application stage – often neglected by research – is where the actual practicality and effectiveness of the model are tested as it involves defining and implementing actions based on the insights derived from the model. The application of a data-driven model can support various types of decision-making: strategic, tactical, or operational. For example, it might involve using the model's output for optimising traffic signal timings, altering public transportation routes, establishing special

lanes, or designing sustainable urban mobility plans. These applications can range from direct use of the model's output to employing it in a secondary modelling process for enhanced decision-making, such as formulating the decision-making process as an optimization problem.

A key aspect of this stage is the model's ability to adapt to real-time changes and support practical decision-making by ITS managers. Techniques such as Stochastic Model Predictive Control (SMPC) exemplify the integration of data-driven models with real-time control methods, efficiently handling complex systems with inherent uncertainties.

5. Adaptation (Iterative Stage)

In the proposed data processing workflow, model adaptation is a critical layer that extends across various stages of the modelling process. Since data-driven models are prone to uncertainties and changes in data patterns, they must be adaptable to maintain accuracy and relevance. This adaptation is essential due to potential changes like variations in traffic flow, new road openings, or unexpected events like public transportation strikes, which can significantly alter user behaviour and, consequently, the data models are based on.

The adaptation process encompasses several stages:

- Preprocessing Stage: Incorporating new data sources, addressing sensor failures, and enhancing data fusion and imputation methods.
- Modelling Stage: Retraining models with new data, switching to alternative models, or modifying learning algorithms.
- Prescription Stage: Adjusting data changes that affect model outputs. Could involve using online learning strategies to quickly adjust to concept drift in data streams.

Adaptations can be either automatic, triggered by certain conditions, or manually introduced based on user inputs. This flexibility enhances the actionability of the model, making it more responsive to the needs of transportation network managers and other end-users. For instance, the introduction of new data sets or the detection of significant data drifts can be managed effectively to ensure the model continues to provide accurate and useful insights.

4.2.4 Coupled Aimsun-FleetPy Simulation Data

4.2.4.1 Introduction

TUM is developing techniques for the efficient integration of urban logistics in DRT services (UC3). The freight and DRT demand data generated by Nommon will be used as input for a co-simulation of Aimsun Next and FleetPy for the city of Madrid. The Madrid network will be provided by Aimsun.

FleetPy is a Python-based DRT simulation tool developed by TUM. It does not have an integrated traffic microsimulation functionality; the travel times are mainly calculated using scaled free-flow travel times obtained from Open Street Map[12] . Thus, it lacks a detailed consideration of other vehicles participating in the overall traffic. To fill this gap, FleetPy is coupled with Aimsun Next using the Python API. The fleet is controlled, i.e., vehicle schedules are computed, in FleetPy while vehicle movements are conducted within the Aimsun environment. The bridge allows the consideration of a

[12] https://www.openstreetmap.org

more realistic traffic simulation in the FleetPy control decisions which replicates the unexpected delays the DRT might face in real traffic.

This section describes the main input data required for the co-simulation of UC3 as well as any further data manipulation done within the co-simulation.

4.2.4.2 Data Used

The main data inputs for UC3 are provided by Nommon and Aimsun and fused together in a co-simulation of FleetPy and Aimsun Next. This will mainly consist of the following:

- A calibrated Aimsun Next network for the Madrid use-case.
- The origin-destination (OD) pair of the DRT passenger request and the time when the request is made.
- The OD pair of the freight requests. If the Madrid use-case is assumed to serve freight requests from a depot, then only the destination (or origins for pickup) points of parcels are required.
- Rest of the FleetPy simulation parameters to describe the fleet control algorithms used.

4.2.4.3 Methodology

For a successful co-simulation of Aimsun Next and FleetPy, some manipulation of the input data is required as described below.

The first and the foremost is the mapping of freight and DRT demand data to the provided network. For simplicity, FleetPy generally limit the locations that can be visited by the DRT fleet to be located on the city network nodes. Thus, instead of the exact geographical locations, the closest node of the city network is used. The pickup or delivery of freight or passengers are, therefore, not considered to be in between network edges, rather, they are assumed to be exactly on network nodes.

The second is in regard to the clustering of the freight requests. The study assumes same-day freight requests, which are significantly less time critical than the DRT passenger requests and can be delivered at any time within the same day. However, it is assumed that the DRT service should at least provide some time window (ranging in hours) within which the freight requests are served. Since it is assumed that the freight requests are mainly last-mile delivery requests known in advance unlike the DRT passenger requests, the freight requests need to be clustered in a way that the estimated freight delivery time windows are fulfilled. Such a clustering can be geographically, temporally or a combination of both. The main challenge faced in this regard is that the service quality of DRT passengers must not be compromised significantly. Therefore, this would require the clustering methods to also consider the DRT passengers' data. Such a clustering of freight requests will be done in a preprocessing step. Figure 28 shows this process.

The third data manipulation is regarding the traffic state data collected from the Aimsun Next simulation. To plan vehicle routes and assign vehicles to DRT requests, FleetPy requires information on the travel times and travel distances between different OD pairs. Since Aimsun Next simulates the traffic on microscopic level, the traffic states and the travel times may change after the assignment of vehicle routes. Nevertheless, FleetPy would require estimating the travel times based on the data collected from Aimsun Next during the simulation. FleetPy solves this problem by considering average travel times on each network edge during a fixed period.

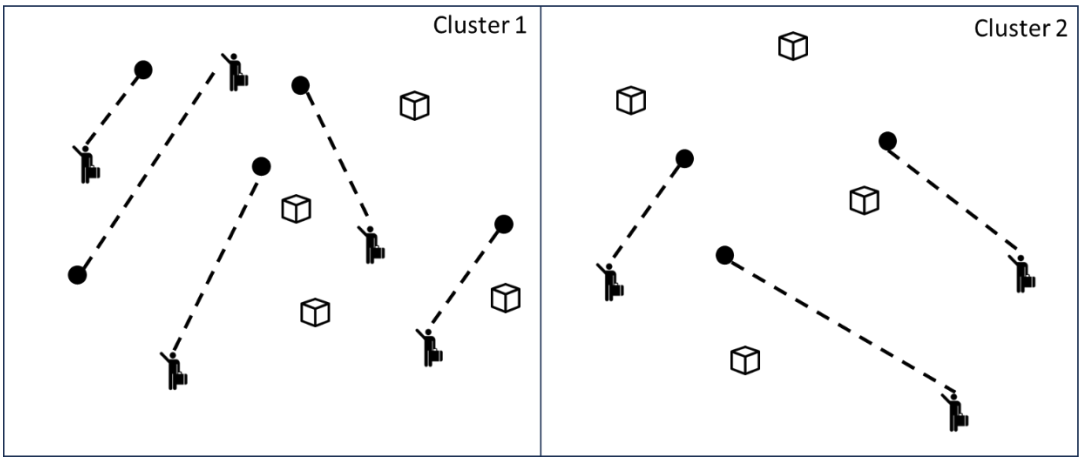

Figure 28 Fusion of DRT passengers and freight data required to create clusters of freight requests.

4.2.5 Space-time Context and Heterogeneous Data Fusion

4.2.5.1 Introduction

This section presents the methodology for data standardization, contextualization and fusion based on graphs proposed by JSI, which will be used in UC2. Data fusion is a common problem in all large-scale enterprise systems. Typically, data are collected and stored in silos. Data in each silo can be stored in a different structure and format. This makes any holistic processing difficult and error prone. The problem can be mitigated by standardizing data format and using explicit contextualization. Doing so allows for explicit data fusion by context which streamlines downstream tasks like data cleaning and feature engineering.

In CONDUCTOR, we are designing a mechanism that explicitly encodes contexts in a graph-based grid-like structure called a Context Graph. The Context Graph discretizes spacetime in a hierarchical grid-like structure that encodes the spatial and temporal relationships between the neighbouring contexts. Each datum is explicitly linked to a context creating an implicit similarity metric and allowing for higher-order reasoning. This type of fusion supports downstream tasks like automatic feature engineering by extracting information from local neighbourhoods and regional graph structure.

The initial prototype is implemented with Neo4J [13] as our graph database. Neo4J offers a standardized query mechanism, simple interactive UI and provides several out-of-the-box graph analytics algorithms and graph embeddings. The current context graph architecture is designed to enable storing the data included in Table 3 in the Context Graph.

Table 3 Data included in the Context Graph

Data Source	Description
Weather Data	The weather data and weather forecasts, for the region of modelling on UC2. The weather data include 5 years of historical weather data. The weather data presents hourly values for key weather parameters,

[13] https://neo4j.com

	including: location long/lat, temperature, wind speed, pressure, humidity, etc.
Flights Information	Flights related data including: time and airport of departure and time and airport of arrival. The data include flights information for all the airports included in GoOpti transport infrastructure. The data includes both historic data since June 2023 and future schedules of up to 1 year in advance. The data presents real-time information on current flight plans and future schedules.
Real-Time Road Infrastructure Data	These data include three main categories of information, namely: (1) Traffic conditions: Road Weather conditions, Static Traffic Events (such as road works), travel times on Motorways, Public Transport Timetable, (2) Traffic infrastructure status: Traffic Border delays, Road Cameras, traffic counters, traffic forecasts, traffic incidents, Wind (real-time measurements on critical sections), (3) Points Of Interests locations: Rest Areas locations, EV Charing infrastructure (chargers' locations), Truck Parkings.
Real-Time Traffic Events (NAP-DARS)	These data includes dynamic events related to road infrastructure. The data are in DATEX II format and includes all status categories related to specific road sections. The basic road events will be augmented with data from border delays estimation models and data from traffic incidents endpoint. The consolidated events data portfolio includes all essential categories that influence route planning and/or estimated time travelled on specific route segment. Such data will be essential for real-time optimization and dynamic routing.

4.2.5.2 Methodology

The designed data fusion methodology, called Context Graph, uses explicit contexts to align and fuse data in a graph. The Context Graph is designed in two layers: (i) the context layer and (ii) the entity layer. Nodes in these layers serve as anchor points that associate each measurement with a context and an entity. The measurement themselves are stored on hyper-edges that connect one or more contexts to one or more entities. The approach allows for several innovative use cases, such as: storing time series in a graph and taking static historic snapshots of the graph.

The context layer forms a grid-like structure that encodes the spatial, temporal and hierarchical relationships between contexts. At a single level of granularity, the context layer forms a 3D grid structure that discretizes spacetime. The idea is illustrated in Figure 29.

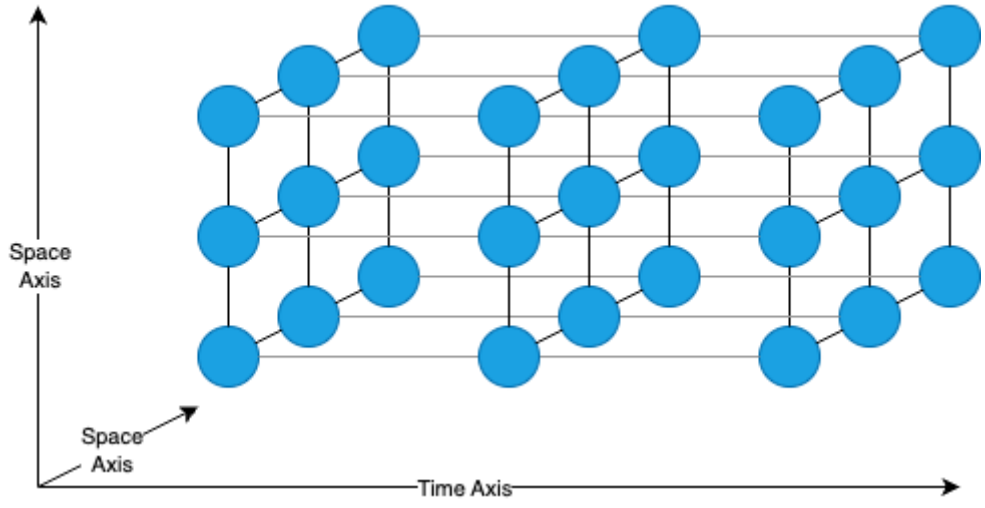

Figure 29 Context layer - 3D grid structure of discretized spacetime.

The formalization of spacetime is presented on Figure 30. The nodes in this structure (e.g. `CityHour`) represent a semantic physical location at a certain point in time. For instance, the blue node represents the city of Ljubljana between 2PM and 3PM at a specific date.

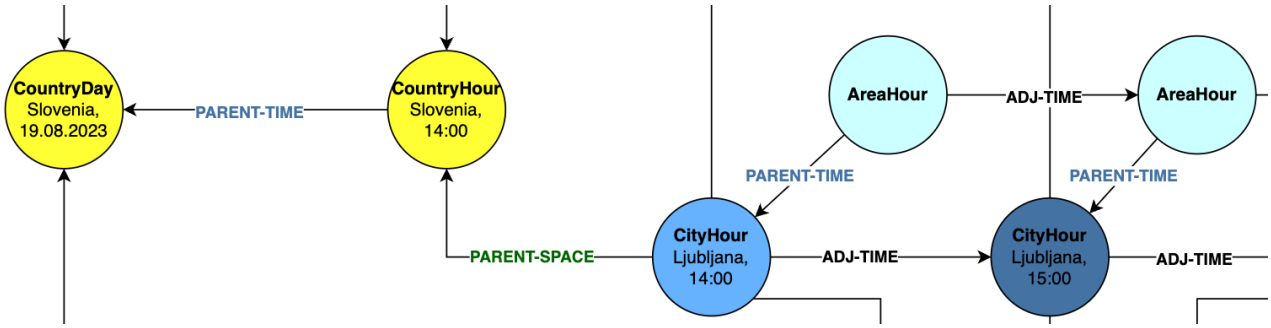

Figure 30 Spacetime formalization of context layers.

Temporal relationships are encoded with `ADJ_TIME` edges while the spatial relationships are encoded with edges labelled `ADJ_SPACE`. Both the spatial and temporal component form a hierarchical structure encoded with edges labelled `PARENT_SPACE` and `PARENT_TIME` respectively. In the example above, we see that node “Ljubljana 14:00” is a child of “Slovenia 14:00”. Figure 31 shows the design of the context layer.

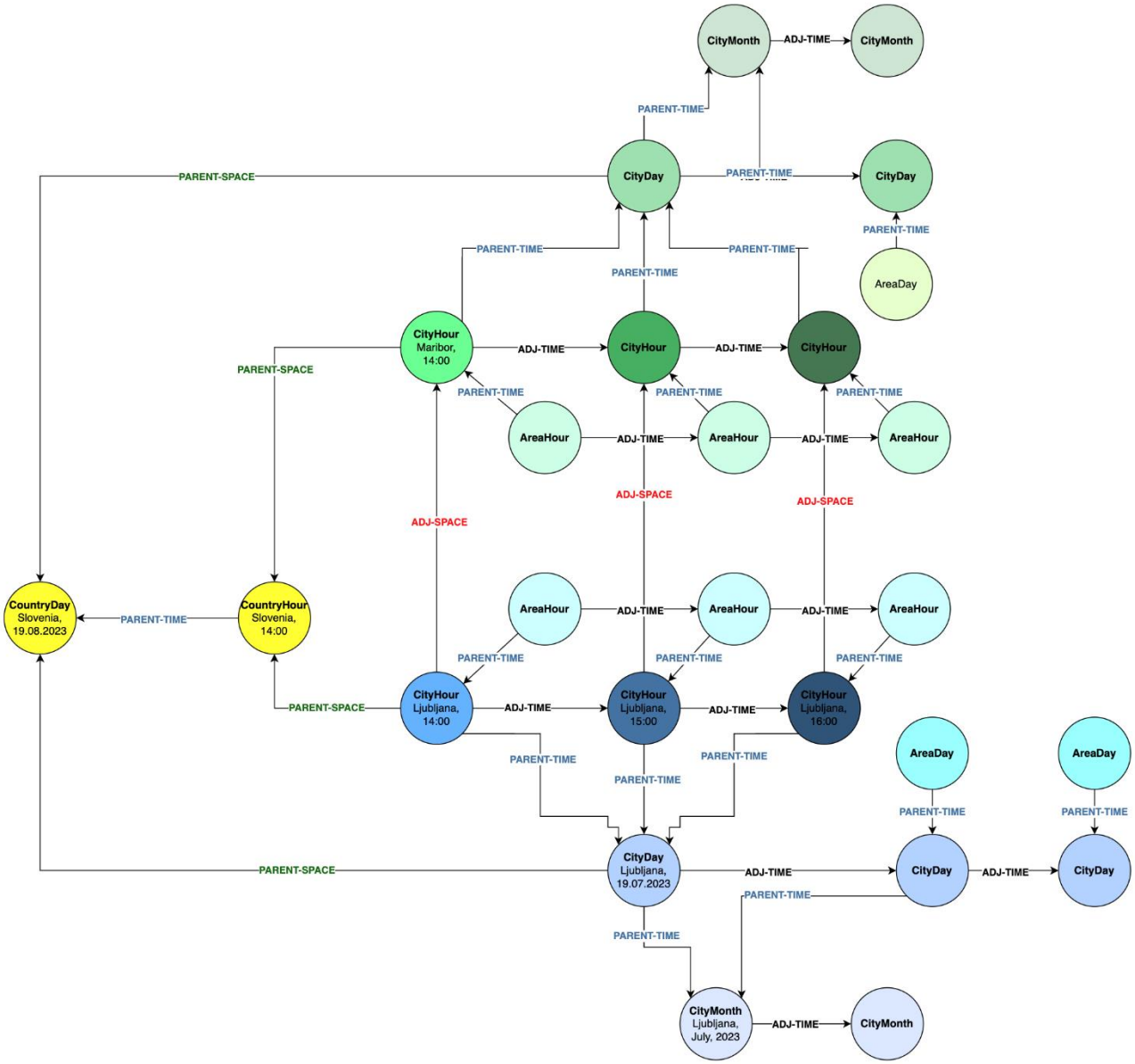

Figure 31 Context layer design.

The entity layer consists of master data and transactional data. In the initial prototype it consists of nodes like `Vehicle`, `Weather` and `TravelOrder` but also entities like holidays and countries. Later it can be extended with semantic structures like ontologies. Figure 32 shows a design of the holiday structure that is used in demand prediction of UC2. In our methodology, we do not restrict edges from the entity layer to the context layer.

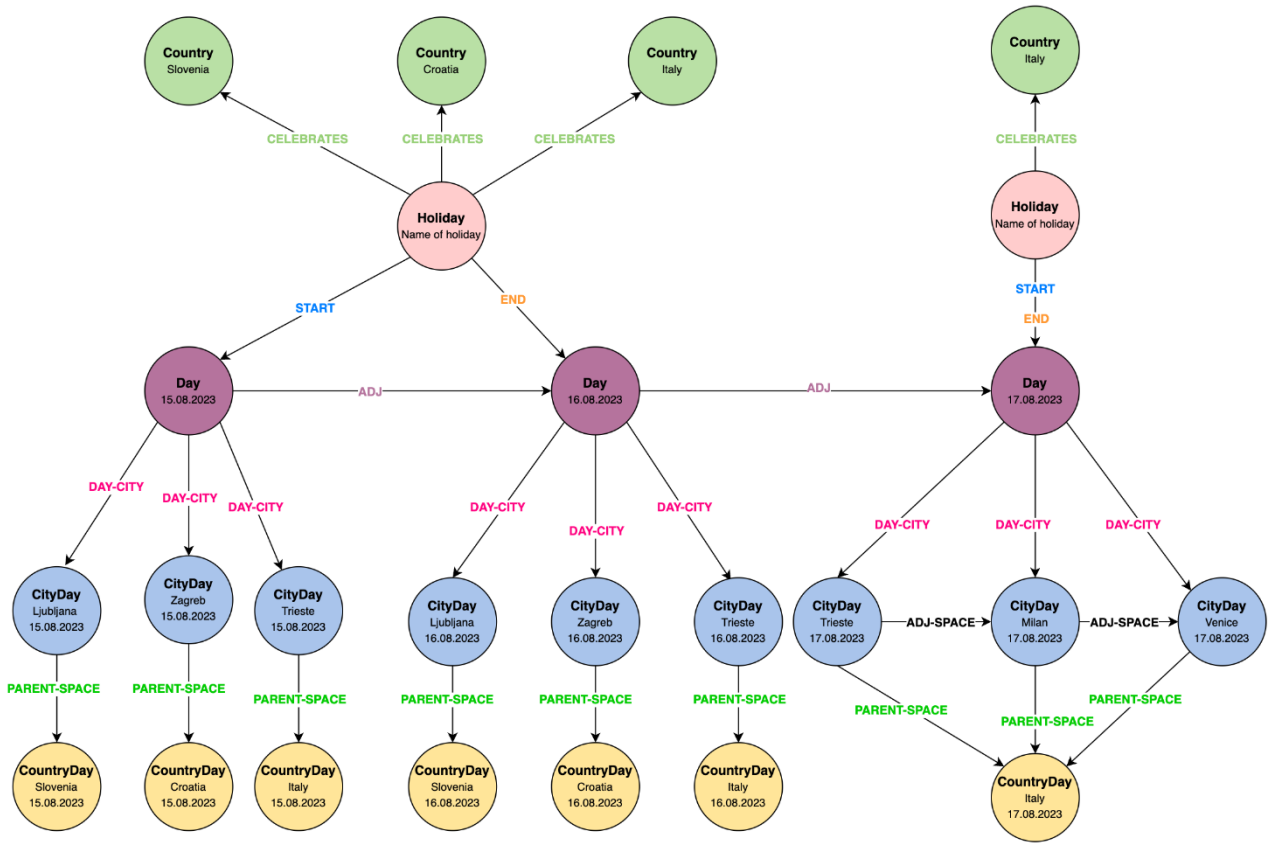

Figure 32 Context layer - design of the holiday structure for demand prediction.

Measurements are stored on a hyperedge that connects a context to one or more entities. For instance, a vehicle speed measurement of 90 km/h taken in Vienna at 1PM, on the 1st of December 2023 is written as `(ch:CityHour)-[:MEASUREMENT {speedKmH: 90}]->(v:Vehicle)` using the Cypher query language. Figure 33 illustrates hourly time-series measurements that are linked to temporally adjacent `CityHour` nodes.

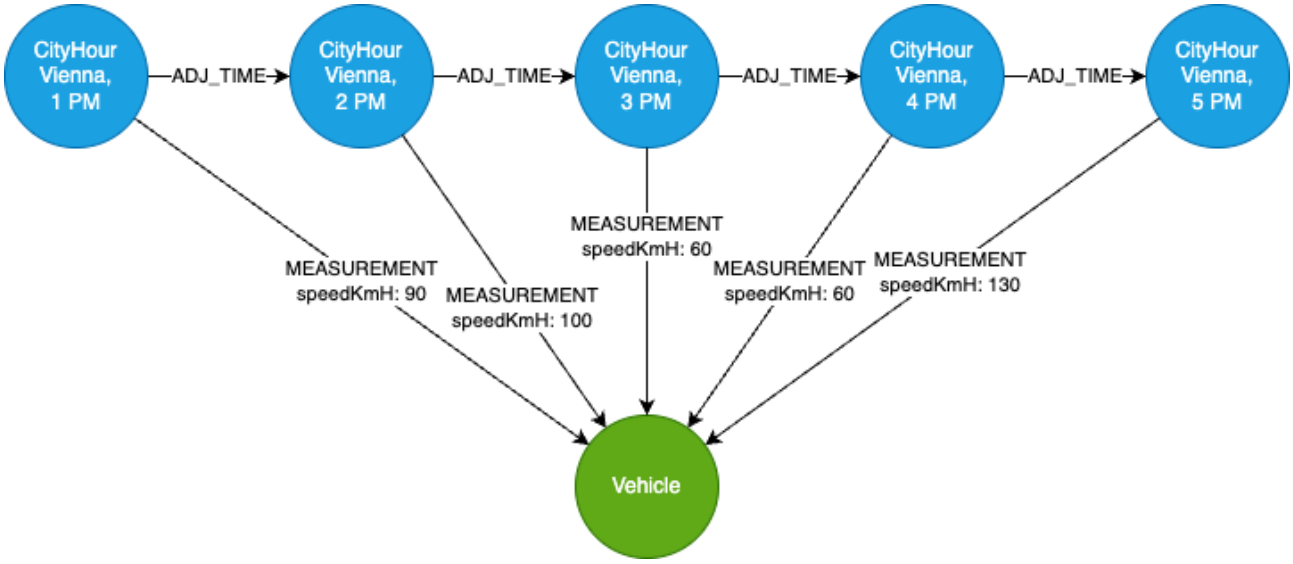

Figure 33 Hyperedge – hourly time-series measurements.

4.2.5.3 Technical Implementation

At the time of writing the Context Graph is implemented as a double-storage architecture. Data is first stored into a single table of a Postgres database and then copied into the Neo4J graph via ETL job. The design allows for easy changes to the graph structure during the research process. The architecture is shown in Figure 34.

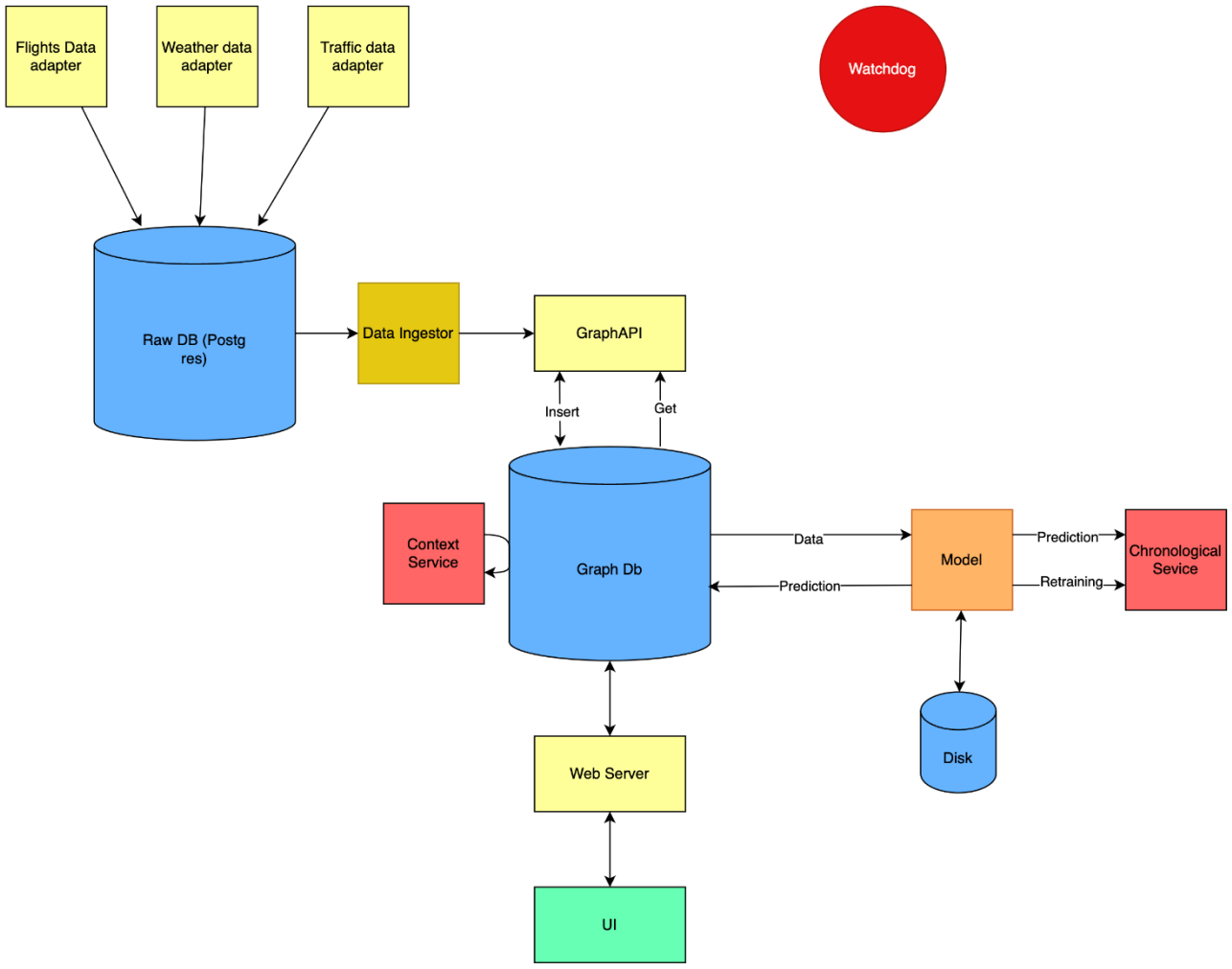

Figure 34 Context Graph architecture.

Each data adapter is implemented as a standalone component. Access to the graph is not provided directly but through a Graph API. The API queues queries to prevent deadlocks that can occur in Neo4J in case of parallel execution. When data is inserted, new contexts are created on-the-fly. At insertion time these are not linked to the grid-like context structure. Instead, they are linked by a nightly job executed by the Context Service. Machine learning models are scheduled using a Cron Service. They read data from the graph and write predictions back into the graph. A web server with an API provides support for user interaction.

4.2.5.4 Validations

The first implementation of the Context Graph still contains sparse data. While the context layer is already connected, the entity layer mostly contains raw entities with little semantic information. Figure 35 shows a sample screenshot of the Context Layer with `Flight` and `CityHour` nodes. At the time of writing, the graph contains 1.4M `Flight` nodes and 2.6M `CityMonth` nodes.

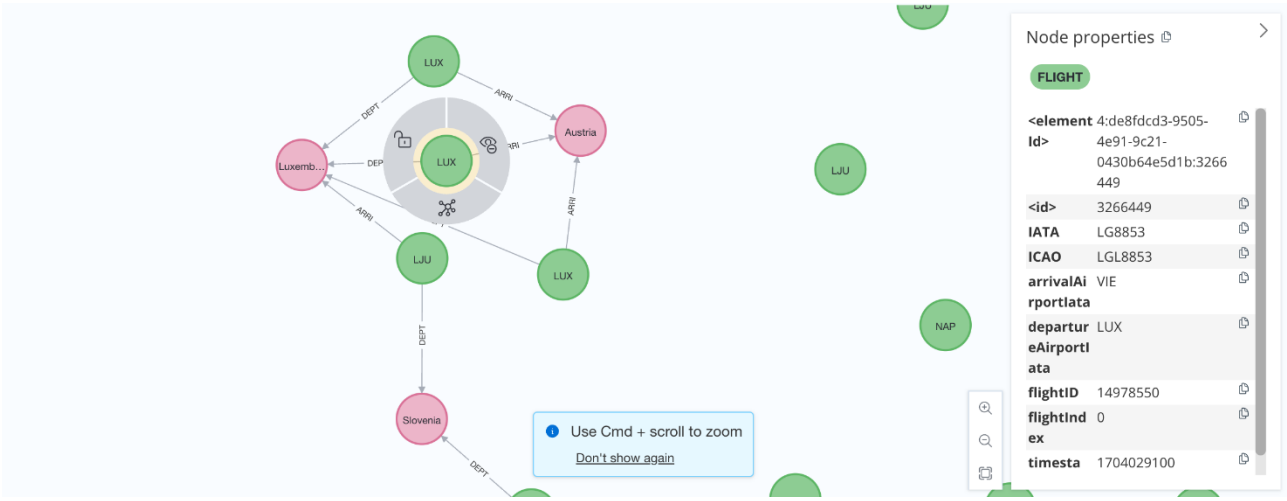

Figure 35 Context layer of Flight and City node.

Figure 36 shows a time series of weather measurements encoded in the Context Graph. In the figure below, we can see both the temporal adjacency structure of the spacetime context (defined by edges `ADJ_TIME_HOUR`) and the hierarchical temporal relationship (defined by edges `PARENT_DAY`). The measurements are stored as properties of the edge `MEASUREMENT`.

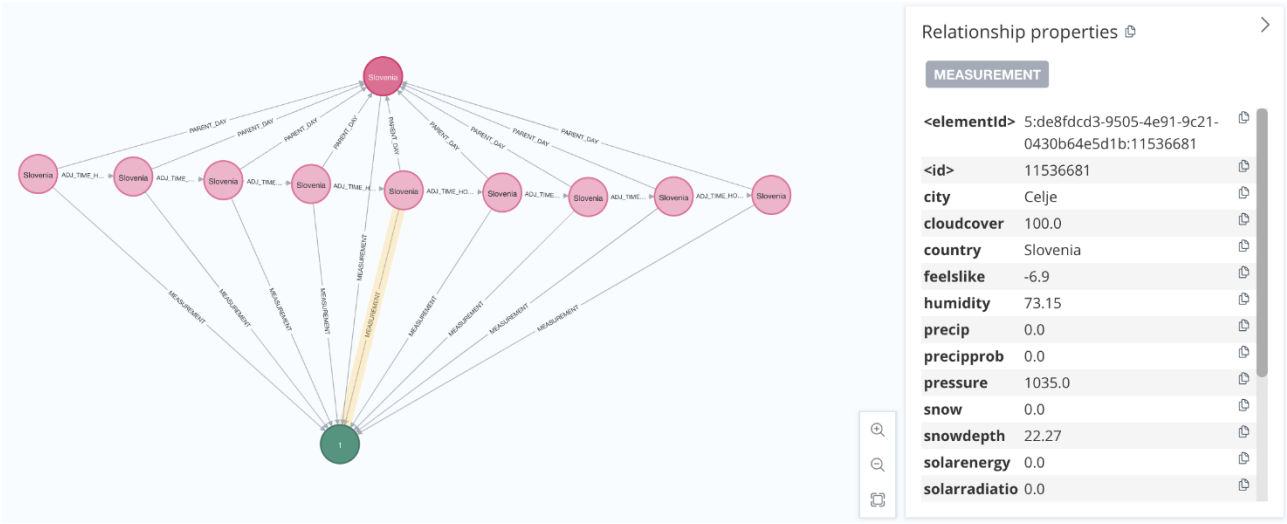

Figure 36 Weather mesurement time series in Context Graph.

During development, we have noticed some pros and cons when compared to a traditional approach of encoding data into relational tables. We summarize these in Table 4.

Table 4 Pros and Cons of Context Graph

Pros	Cons
Easier search and traversal of contextually similar data. Data with similar context can be identified by traversing the graph structure as opposed to manual identification of similar context with relational databases. For instance, by expanding two or three nodes in succession one can identify flights that landed on neighbouring airports.	Increased effort for data modelling and query design. We found that arbitrary queries on a graph run slower than on traditional databases. Additional effort needs to be placed into data modelling, indexing and query design.

Availability of out-of-the-box analytical algorithms. Graph algorithms like shortest-path and page rank can be used to define metrics like similarity and centrality. These are not available in relational databases.	Handling concurrency and deadlocking. When performing write operations one needs to be careful of parallel execution. Because many nodes are linked, writing and editing data requires locking which can slow down query performance drastically and even lead to deadlocks. To mitigate, we placed an API layer on top of the graph that ensures that queries are executed sequentially.

In the future, the approach will be rigorously validated before focusing on data enrichment and experimentation with automatic feature extraction. The validation will include stress tests and query performance evaluation where we see the highest risk. If successful, we will experiment with automatic feature extraction with graph analysis algorithms (i.e., connected features) and graph embeddings. The primary future directions are semantic data enrichment with ontologies and automatic machine learning.

5 CONCLUSIONS

The functionalities designed and developed in this report will be implemented and tested across five pilot projects and validated on three different use cases. This deliverable presents the initial version of the design, methods and specification for data gathering harmonization and data fusion and analysis, as a result of the ongoing Tasks 3.1 and 3.2.

Moreover, the FIWARE's smart data models were selected as the basis for harmonisation within CONDUCTOR. The use of FIWARE's smart data models for data harmonisation in CONDUCTOR is based on their suitability and alignment with the project's goals. As presented in Chapter 3, the common data model was used as a framework to design Context Broker that will enable to manage the entire lifecycle of context information including updates, queries, registrations and subscriptions. Using the Context Broker, one is able to create context elements and manage them through updates and queries. In such a way, the data integration and data management are enabled on semantic level.

By adopting common information models for data representation and applying data space design with Context Broker and big data architecture deployment, CONDUCTOR can ensure seamless integration of applications and enable efficient investigation of CCAM services. The harmonised representation of data models will enable easier sharing and exchange of information among different components of the project.

As part of the data fusion tasks of CONDUCTOR, five developments were identified as relevant for the design of new traffic management strategies:

- Characterisation of delivery trips and estimation of delivery demand from mobile network, surveys and logistic operation data.
- Identification of unusual traffic patterns caused by large-scale events.
- Framework for actionable smartphone-based data analytics.
- FleetPy—Aimsun coupling specification.
- Space-time context and heterogeneous data fusion

The methodology of all of them is presented in detail, as well as the initial implementation steps. The developments will be applied and validated in the different UCs of the project, and, as UCs progress, they will be refined.

As mentioned in Section 4, even though each development is framed within one of the CONDUCTOR UCs, during the definition and implementation phases, a from-particular-to-general approach is being followed, in which each development allows the definition of general methodologies that can be extrapolated. So, the same development can potentially be applied to more than one UC.

The report presents the initial version of methods, designs and specifications for final data integration. It will be updated, tested and validated on the designed UCs and pilot setups and the final designs reported in final reporting.

6 REFERENCES

Adamidis, F. K., Mantouka, E. G., & Vlahogianni, E. I. (2020). Effects of controlling aggressive driving behavior on network-wide traffic flow and emissions. International Journal of Transportation Science and Technology, 9(3), 263–276. https://doi.org/10.1016/j.ijtst.2020.05.003

Afyouni, I., Khan, A., & Al Aghbari, Z. (2022). Deep-Eware: spatio-temporal social event detection using a hybrid learning model. Journal of Big Data 9, 86.

Afyouni, I., Khan, A., & Al Aghbari, Z. (2023). E-ware: a big data system for the incremental discovery of spatio-temporal events from microblogs. Journal of Ambient Intelligence and Humanized Computing 14, 13949-13968.

Bejani, M. M., & Ghatee, M. (2018). A context aware system for driving style evaluation by an ensemble learning on smartphone sensors data. Transportation Research Part C: Emerging Technologies, 89, 303–320. https://doi.org/10.1016/j.trc.2018.02.009

Belcastro, L., Marozzo, F., Talia, D., Trunfio, P., Branda, F., Palpanas T., & Imran, M. (2021). Using social media for sub-event detection during disasters. Journal of Big Data 8, 79.

Bertossi, L., & Geerts, F. (2020). Data quality and explainable AI. ACM Journal of Data Management and Information Quality 12(2), 1-9.

Campolina, A., Boukerche, A., & Loureiro, A. A. (2020). Context and location awareness in eco-driving recommendations. 1–6.

Christin, D. (2016). Privacy in mobile participatory sensing: Current trends and future challenges. Journal of Systems and Software, 116, 57–68. https://doi.org/10.1016/j.jss.2015.03.067

CONDUCTOR Consortium (2023). D1.2 Specification of the future mobility system and data sources. CONDUCTOR project, Deliverable D1.0. Version 1.0. April 2023.

Etemad, M., Soares Júnior, A., & Matwin, S. (2018). Predicting Transportation Modes of GPS Trajectories Using Feature Engineering and Noise Removal. In E. Bagheri & J. C. K. Cheung (Eds.), Advances in Artificial Intelligence (pp. 259–264). Springer International Publishing. https://doi.org/10.1007/978-3-319-89656-4_24

Fafoutellis, P., Mantouka, E. G., & Vlahogianni, E. I. (2020). Eco-driving and its impacts on fuel efficiency: An overview of technologies and data-driven methods. Sustainability, 13(1), 226.

Fafoutellis, P., Mantouka, E. G., & Vlahogianni, E. I. (2022). Acceptance of a Pay-How-You-Drive pricing scheme for city traffic: The case of Athens. Transportation Research Part A: Policy and Practice, 156, 270–284.

Fafoutellis, P., Mantouka, E. G., Vlahogianni, E. I., & Fortsakis, P. (2023). Investigating the impacts of the COVID-19 pandemic on Eco-driving behavior. Safety Science, 166, 106251. https://doi.org/10.1016/j.ssci.2023.106251

Ferreira da Silva, F.-F., Duarte, J.-C., & Ugulino, W.-C. (2022). Automated statistics extraction of public security events reported through microtexts on social networks. SBSI ’22: Proceedings of the XVIII Brazilian Symposium on Information Systems, pp. 1-7.

Geneshkumar, P., Ashwin Kumar, B. R., Padmanabhan, S., & Vetriselvi, A. (2022). Social media personal event notifier using NLP and deep learning. Proceedings of the 2022 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), pp. 1-5.

Gilman, E., Keskinarkaus, A., Tamminen, S., Pirttikangas, S., Röning, J., & Riekki, J. (2015). Personalised assistance for fuel-efficient driving. Transportation Research Part C: Emerging Technologies, 58, 681–705.

Gunning, D., Stefik, M., Choi, J., Miller, T., Stumpf, S., & Yang, G.-Z. (2019). XAI – Explainable artificial intelligence. Science Robotics 4(37), 1-2.

Handel, P., Skog, I., Wahlstrom, J., Bonawiede, F., Welch, R., Ohlsson, J., & Ohlsson, M. (2014). Insurance telematics: Opportunities and challenges with the smartphone solution. IEEE Intelligent Transportation Systems Magazine, 6(4), 57–70.

Hinton, G., & Roweis, S. (2002). Stochastic neighbor embedding. Proceedings of the 2002 Conference and Workshop on Neural Information Processing Systems.

Hodorog, A., Petri, I., & Rezgui, Y. (2022). Machine learning and natural language processing of social media data for event detection in smart cities. Sustainable Cities and Society 85, 104026.

Hu, X. L., Zhang, L. C., & Wang, Z. X. (2018). An adaptive smartphone anomaly detection model based on data mining. EURASIP Journal on Wireless Communications and Networking, 2018(1), 148. https://doi.org/10.1186/s13638-018-1158-6

Jay, J., Heykoop, F., Hwang, L., Courtepatte, A., de Jong, J., & Kondo, M. (2022). Use of smartphone mobility data to analyze city park visits during the COVID-19 pandemic. Landscape and Urban Planning, 228, 104554. https://doi.org/10.1016/j.landurbplan.2022.104554

Johnson, D. A., & Trivedi, M. M. (2011). Driving style recognition using a smartphone as a sensor platform. 1609–1615.

Khaleghi, B., Khamis, A., Karray, F., & Razavi, S. (2013). Multisensor data fusion: A review of the state-of-the-art. Information Fusion 14, 28-44.

Konstantinou, C., Fafoutellis, P., Mantouka, E. G., Chalkiadakis, C., Fortsakis, P., & Vlahogianni, E. I. (2023). Effects of Driving Behavior on Fuel Consumption with Explainable Gradient Boosting Decision Trees. 1–6.

Krieg, J.-G., Jakllari, G., Toma, H., & Beylot, A.-L. (2018). Unlocking the smartphone's sensors for smart city parking. Pervasive and Mobile Computing, 43, 78–95. https://doi.org/10.1016/j.pmcj.2017.12.002

Laña, I., Sanchez-Medina, J. J., Vlahogianni, E. I., & Del Ser, J. (2021). From Data to Actions in Intelligent Transportation Systems: A Prescription of Functional Requirements for Model Actionability. Sensors, 21(4), Article 4. https://doi.org/10.3390/s21041121

Liu, F. T., Ting, K. M., & Zhou, H. (2008). Isolation Forest. Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy.

van der Maaten, L. J. P., & Hinton, G. E. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research 9*, 2579-2605.

Magaña, V. C., & Organero, M. M. (2014). The impact of using gamification on the eco-driving learning. 45–52.

Maldonado, S., & López, J. (2018). Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification. Applied Soft Computing, 67, 94–105. https://doi.org/10.1016/j.asoc.2018.02.051

Mantouka, E. G., Fafoutellis, P., & Vlahogianni, E. I. (2021). Deep survival analysis of searching for on-street parking in urban areas. Transportation Research Part C: Emerging Technologies, 128, 103173. https://doi.org/10.1016/j.trc.2021.103173

Mantouka, E. G., & Vlahogianni, E. I. (2022). Deep Reinforcement Learning for Personalized Driving Recommendations to Mitigate Aggressiveness and Riskiness: Modeling and Impact Assessment. Transportation Research Part C: Emerging Technologies, 142, 103770. https://doi.org/10.1016/j.trc.2022.103770

McInnes, L., & Healy, J. (2018). UMAP: Uniform Manifold Approximation and Projection for Dimensional Reduction. ArXiv e-prints 1802.03426.

Meng, T., Jing, X., Yan, Z., & Pedrycz, W. (2020). A survey on machine learning for data fusion. Information Fusion 57, 115-129.

Mitchell, H. B. (2012). Data fusion: Concepts and ideas. New York, United States of America: Springer.

Mourtakos, V., Mantouka, E. G., Fafoutellis, P., Vlahogianni, E. I., & Kepaptsoglou, K. (2023). Reconstructing mobility from smartphone data: Empirical evidence of the effects of COVID-19 pandemic crisis on working and leisure. Transport Policy. https://doi.org/10.1016/j.tranpol.2023.11.018

Niemiec, M. (2017). *Fuzzy Inference System: Theory and Applications*. Scitus Academics LLC.

Nitsche, P., Widhalm, P., Breuss, S., Brändle, N., & Maurer, P. (2014). Supporting large-scale travel surveys with smartphones – A practical approach. Transportation Research Part C: Emerging Technologies, 43, 212–221. https://doi.org/10.1016/j.trc.2013.11.005

Predic, B., & Stojanovic, D. (2015). Enhancing driver situational awareness through crowd intelligence. Expert Systems with Applications, 42(11), 4892–4909.

Rashinkar, P., & Krushnasamv, V. S. (2017). An overview of data fusion techniques. Proceedings of the 2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), pp. 694-697.

Roy, A., Cruz, R. M. O., Sabourin, R., & Cavalcanti, G. D. C. (2018). A study on combining dynamic selection and data preprocessing for imbalance learning. Neurocomputing, 286, 179–192. https://doi.org/10.1016/j.neucom.2018.01.060

Salpietro, R., Bedogni, L., Di Felice, M., & Bononi, L. (2015). Park Here! A smart parking system based on smartphones' embedded sensors and short range Communication Technologies. 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), 18–23. https://doi.org/10.1109/WF-IoT.2015.7389020

Servizi, V., Pereira, F. C., Anderson, M. K., & Nielsen, O. A. (2021). Transport behavior-mining from smartphones: A review. European Transport Research Review, 13(1), 57. https://doi.org/10.1186/s12544-021-00516-z

Shukla, A. K., Sharma, R., & Muhuri, P. K. (2018). A Review of the Scopes and Challenges of the Modern Real-Time Operating Systems. International Journal of Embedded and Real-Time Communication Systems (IJERTCS), 9(1), 66–82. https://doi.org/10.4018/IJERTCS.2018010104

Spolaor, S., Fuchs, C., Cazzaniga, P., Kaymak, U., Bessozi, D., & Nobile, M. (2020). Simpful: A User-Friendly Python Library for Fuzzy Logic. *International Journal of Computational Intelligence Systems 13*(1), 1687-1698.

Stavrakaki, A.-M., Tselentis, D. I., Barmpounakis, E., Vlahogianni, E. I., & Yannis, G. (2020). Estimating the Necessary Amount of Driving Data for Assessing Driving Behavior. Sensors, 20(9), Article 9. https://doi.org/10.3390/s20092600

Stopher, P. R., & Greaves, S. P. (2007). Household travel surveys: Where are we going? Transportation Research Part A: Policy and Practice, 41(5), 367–381.

Thomas, T., Geurs, K. T., Koolwaaij, J., & Bijlsma, M. (2018). Automatic Trip Detection with the Dutch Mobile Mobility Panel: Towards Reliable Multiple-Week Trip Registration for Large Samples. Journal of Urban Technology, 25(2), 143–161. https://doi.org/10.1080/10630732.2018.1471874

Tselentis, D. I., Vlahogianni, E. I., & Yannis, G. (2019). Driving safety efficiency benchmarking using smartphone data. Transportation Research Part C: Emerging Technologies, 109, 343–357.

Tselentis, D. I., Yannis, G., & Vlahogianni, E. I. (2017). Innovative motor insurance schemes: A review of current practices and emerging challenges. Accident Analysis & Prevention, 98, 139–148.

Vitanza, E., Dimitri, G.-M., & Mocenni, C. (2023). A multi-modal machine learning approach to detect extreme rainfall events in Sicily. Scientific Reports 13, 6196.

Vlahogianni, E. I., & Barmpounakis, E. N. (2017a). Driving analytics using smartphones: Algorithms, comparisons and challenges. Transportation Research Part C: Emerging Technologies, 79, 196–206. https://doi.org/10.1016/j.trc.2017.03.014

Vlahogianni, E. I., & Barmpounakis, E. N. (2017b). Gamification and sustainable mobility: Challenges and opportunities in a changing transportation landscape.

Vlahogianni, E. I., Yannis, G., & Golias, J. C. (2013). Critical power two wheeler driving patterns at the emergence of an incident. Accident Analysis & Prevention, 58, 340–345.

Vlahogianni, E. I., Yannis, G., & Golias, J. C. (2014). Detecting powered-two-wheeler incidents from high resolution naturalistic data. Transportation Research Part F: Traffic Psychology and Behaviour, 22, 86–95.

Wahlström, J., Skog, I., & Händel, P. (2015). Detection of dangerous cornering in GNSS-data-driven insurance telematics. IEEE Transactions on Intelligent Transportation Systems, 16(6), 3073–3083.

Wang, J., Wang, Y., Zhang, D., & Helal, S. (2018). Energy Saving Techniques in Mobile Crowd Sensing: Current State and Future Opportunities. IEEE Communications Magazine, 56(5), 164–169. https://doi.org/10.1109/MCOM.2018.1700644

Wang, L., Gjoreski, H., Ciliberto, M., Mekki, S., Valentin, S., & Roggen, D. (2019). Enabling Reproducible Research in Sensor-Based Transportation Mode Recognition With the Sussex-Huawei Dataset. IEEE Access, 7, 10870–10891. https://doi.org/10.1109/ACCESS.2019.2890793

White, J., Thompson, C., Turner, H., Dougherty, B., & Schmidt, D. C. (2011). Wreckwatch: Automatic traffic accident detection and notification with smartphones. Mobile Networks and Applications, 16, 285–303.

Yen, B. T., Mulley, C., & Burke, M. (2019). Gamification in transport interventions: Another way to improve travel behavioural change. Cities, 85, 140–149.

Zhao, F., Ghorpade, A., Pereira, F. C., Zegras, C., & Ben-Akiva, M. (2015). Stop Detection in Smartphone-based Travel Surveys. Transportation Research Procedia, 11, 218–226. https://doi.org/10.1016/j.trpro.2015.12.019

Zhao, F., Pereira, F. C., Ball, R., Kim, Y., Han, Y., Zegras, C., & Ben-Akiva, M. (2015). Exploratory analysis of a smartphone-based travel survey in Singapore. Transportation Research Record, 2494(1), 45–56.

Zhu, C., Han, B., & Zhao, Y. (2020). A Comparative Study of Spark on the bare metal and Kubernetes. Proceedings - 2020 6th International Conference on Big Data and Information Analytics, BigDIA 2020, 117–124. https://doi.org/10.1109/BigDIA51454.2020.00027

A. APPENDIX

Demo site	Data category	Data type	Data source	Format
Athens	Environmental conditions	Weather	Weather Underground	
Athens	Environmental conditions	Weather	National Observatory of Athens (NOA)	Plain text
Athens, Madrid, Almelo, Slovenia	Environmental conditions	Weather	Visual Crossing	JSON, CSV, Excel XLSX
Athens, Madrid, Almelo, Slovenia	Environmental conditions	Weather	OpenWeatherMap	JSON, XML, HTML
Athens, Madrid, Almelo, Slovenia	Environmental conditions	Air pollution	OpenWeatherMap	JSON
Athens	Demographics	Census data	Hellenic Statistical Authority (ELSTAT)	Excel XLSX, Plain text
Athens	Traffic conditions	PT bus/metro ridership	OASA	JSON, CSV
Athens	Traffic conditions	PT bus/metro ridership	OASA	
Athens	Traffic conditions	Telematics data	OASA	CSV
Athens	Traffic conditions	Telematics data	OASA	JSON
Athens	Transport supply, PT offer	PT bus routes and stations	OASA (OSY)	JSON, CSV, XML, TSV
Athens	Transport supply, PT offer	PT metro routes and stations	OASA (STASY)	JSON, CSV, XML, TSV
Athens	Traffic conditions	Road traffic (loop detector data)	Region of Attica	JSON, CSV

Madrid	Environment conditions	Weather	Spanish Weather Agency (Agencia Española de Meteorología - AEMET)	JSON
Madrid	Demographics	Census data	Spanish National Statistical Office (INE)	CSV, Excel XLSX, JSON, TSV
Madrid	Surveys	Mobility Household Survey	Regional Transport Authority/ Community of Madrid	Excel XLSX
Madrid		Land Use data	Spanish National Geographic Information Centre (CNIG)	
Madrid	Transport supply, PT offer	PT infrastructure data (stations and lines for suburban train, metro, intercity bus)	Statistics Institute of the Community of Madrid	
Madrid	Transport supply, PT offer	PT schedules data	Madrid Regional Transport Consortium (CRTM)	CSV, GTFS
Madrid		Mobile Network Operator (MNO) data	One of the largest telecom companies in Spain	Plain text, ZIP
Madrid	Surveys	E-commerce survey data	Spanish National Statistical Office (INE)	JSON, CSV, Excel XLSX
Madrid	Surveys	Contact with new technologies survey data	Spanish National Statistical Office (INE)	JSON, CSV, Excel XLSX
Madrid	Traffic conditions	Shared mobility vehicle location data	Fluctuo	JSON, CSV
Madrid	Road network model	Madrid M-30 network model	Aimsun Next	ANG
Madrid	Transport demand, Generated data	OD matrices	Data Fusion for Travel Demand Estimation and Characterisation	CSV

Slovenia	Environmental conditions	Weather	WeatherAPI	JSON, XML
Slovenia	Environmental conditions	Weather (forecast)	WeatherAPI	JSON, XML
Slovenia	Road network	Physical infrastructure	OpenStreetMap (OSM)	OSM PBF, Shapefile
Slovenia	Road network	DARS infrastructure data	National Access Point (NAP)	JSON, XML, DATEX II (XML)
Slovenia	Traffic conditions	DARS infrastructure data	Traffic information centre (Promet)	CSV, Excel XLSX
Slovenia	Transport demand	Pickup-drop-off data	GoOpti	
Slovenia	Transport demand	Flight information data	OAG	JSON
Slovenia	Transport demand, Generated data	Traffic demand prediction	DRT Demand Prediction	
Almelo	Environmental conditions	Weather	Royal Netherlands Meteorological Institute (KNMI)	netCDF, Plain text
Almelo	Environmental conditions	Weather	Buienradar	JSON
Almelo	Traffic conditions	Radar/CCTV data	Municipality of Almelo	
Almelo	Traffic conditions	Roadside sensors data	Municipality of Almelo	V-Log
Almelo	Traffic conditions	Fleet data	Logistic companies	
Almelo	Transport supply, PT offer	PT routes and stops	Open Mobility Data	GTFS, GTFS-RT

B. ABBREVIATIONS AND DEFINITIONS

AI	Artificial Intelligence
ADAS	Advanced Driving Assistance Systems
API	Application Programming Interface
CCAM	Connected, Cooperative and Automated Mobility
CI/CD	Continuous Integration/Continuous Delivery
CDR	Call Detail Record
DRT	Demand-Responsive Transport
ETL	Extract Transform Load
EV	Electric Vehicle
GPS	Global Positioning System
IDSA	International Data Spaces Association
INE	Spanish National Statistics Institute
ITS	Intelligent Transport System
ML	Machine Learning
MND	Mobile Network Data
MVD	Minimum Viable Dataspace
OD	Origin-Destination
POI	Point Of Interest
SMPC	Stochastic Model Predictive Control
UC	Use Case