MOBIDATALAB

Labs for prototyping future mobility data sharing solutions in the cloud

D4.2 Transport Cloud Architecture Dossier V2

09/02/2023

Author(s): Salvatore TRANI (CNR), Francesco LETTICH (CNR), Alberto BLANCO JUSTICIA (URV), Sorel SIGHOKO (AKKODIS), Renée OBREGON GONZALEZ (AKKODIS), Thierry CHEVALLIER (AKKODIS), Didier de RYCK (HOVE), Victor LEPAGE (HOVE), Johannes LAUER (HERE)

MobiDataLab is funded by the EU under the H2020 Research and Innovation Programme (grant agreement No 101006879).

Summary sheet

Deliverable Number	D4.2
Deliverable Name	Transport Cloud Architecture dossier V2
Full Project Title	MobiDataLab, Labs for prototyping future Mobility Data sharing cloud solutions
Responsible Author(s)	Salvatore TRANI (CNR)
Contributing Partner(s)	CNR, AKKODIS, HOVE, HERE, URV
Peer Review	ICOOR, AETHON
Contractual Delivery Date	31-01-2023
Actual Delivery Date	31-01-2023
Status	Final
Dissemination level	Public
Version	V1.0
No. of Pages	58
WP/Task related to the deliverable	WP4 / T4.1
WP/Task responsible	AKKODIS / CNR
Document ID	MobiDataLab-D4.2-TransportCloudArchitectureDossierV2-v1.0
Abstract	This deliverable provides an overview of the results of Task 4.1, which consists of the conception, definition and design of the transport cloud solution supporting the data federation to enable the European-wide data sharing and trans-national access to the information provided by the federated repositories. This deliverable represents an evolution of the deliverable D4.1, whereby the reviews, learnings, and results gathered between M12 and M24 on work package 4 related tasks were integrated.

Legal Disclaimer

MOBIDATALAB (Grant Agreement No 101006879) is a Research and Innovation Actions project funded by the EU Framework Programme for Research and Innovation Horizon 2020. This document contains information on MOBIDATALAB core activities, findings, and outcomes. The content of this publication is the sole responsibility of the MOBIDATALAB consortium and cannot be considered to reflect the views of the European Commission.

Project partners

Organisation	Country	Abbreviation
AKKODIS	France	AKKODIS
CONSORZIO INTERUNIVERSITARIO PER L'OTTIMIZZAZIONE E LA RICERCA OPERATIVA	Italy	ICOOR
AETHON SYMVOULI MICHANIKI MONOPROSOPI IKE	Greece	AETHON
CONSIGLIO NAZIONALE DELLE RICERCHE	Italy	CNR
HOVE	France	HOVE
HERE GLOBAL B.V.	Netherlands	HERE
KATHOLIEKE UNIVERSITEIT LEUVEN	Belgium	KUL
UNIVERSITAT ROVIRA I VIRGILI	Spain	URV
POLIS - PROMOTION OF OPERATIONAL LINKS WITH INTEGRATED SERVICES	Belgium	POLIS
F6S NETWORK IRELAND LIMITED	Ireland	F6S

Funded by the European Union

Document history

Version	Date	Organisation	Main area of changes	Comments
0.2	27/10/2022	CNR	All	Definition of the TOC
0.4	09/12/2022	CNR, AKKODIS, URV, HOVE, HERE	All	Collection contributions from the partner
0.6	02/01/2023	CNR	All	Document Consolidated
0.8	17/01/2023	AETHON, ICOOR, AKKODIS	All	Internal Review
0.9	20/01/2023	CNR	All	Rework including internal review comments
1.0	30-31/01/2023	AKKODIS	All	Quality check and submission

Executive Summary

The Deliverable D4.2 is the second submitted version of the document that describes the components and architecture of the MobiDataLab Transport Cloud. The deliverable provides the design principles and technical guidelines that allow the platform functionalities to be aligned with the requirements of the stakeholders (i.e., authorities, operators, MaaS companies, and developers) represented in the project.

More specifically, the document specifies how the various non-functional, functional, and general requirements are implemented, it provides an overview on the technologies that have been selected for the implementation of the Transport Cloud Architecture, and finally it allows to define an update on the status of the most promising initiatives related to MobiDataLab.

Table of contents

List of figures

List of tables

Abbreviations and acronyms

Abbreviation	Meaning
ACID	Atomicity, Consistency, Isolation, Durability
API	Application Programming Interface
CAS	Central Authentication Service
CKAN	Comprehensive Knowledge Archive Network
CLI	Command Line Interface
CSV	Comma-Separated Values
CSW	Catalogue Service for the Web

DCAT	Data Catalogue Vocabulary
DOS	Denial of Service
ETA	Estimated Time of Arrival
GIS	Geographic Information System
GTFS	General Transit Feed Specification
GUI	Graphical User Interface
HDFS	Hadoop Distributed File System
HTTP	Hypertext Transfer Protocol
IAAS	Infrastructure as a Service
IAM	Identity and Access Management
INSPIRE	Infrastructure for Spatial Information in the European Community
JS	JavaScript
JSON	JavaScript Object Notation
LDAP	Lightweight Directory Access Protocol
MaaS	Mobility as a Service
OAuth	Open Authorisation
OGC	Open Geospatial Consortium
OSM	OpenStreetMap
OWL	Web Ontology Language
PBF	Protocol buffer Binary Format
PDF	Portable Document Format
POI	Point of Interest
RDF	Resource Description Framework

REST	Representational State Transfer
RPM	Red Hat Package Manager
SAML	Security Assertion Markup Language
SCIM	System for Cross-domain Identity Management
SIRI	Service Interface for Real-time Information
SPARQL	SPARQL Protocol and RDF Query Language
SPML	Service Provisioning Markup language
SSO	Single Sign-On
UMA	User-Managed Access
XACML	eXtensible Access Control Markup Language
XML	eXtensible Markup Language

Funded by the European Union

1. Introduction

1.1. Project overview

There has been an explosion of mobility services and data sharing in recent years. Building on this, the EU-funded MobiDataLab project works to foster the sharing of data amongst transport authorities, operators and other mobility stakeholders in Europe. MobiDataLab develops knowledge as well as a cloud solution aimed at easing the sharing of data. Specifically, the project is based on a continuous co-development of knowledge and technical solutions. It collects and analyses the advice and recommendations of experts and supporting cities, regions, clusters and associations. These actions are assisted by the incremental construction of a cross-thematic knowledge base and a cloud-based service platform, which will improve access and usage of data sharing resources.

1.2. Purpose of the deliverable

The purpose of this document, i.e., the “Transport Cloud Architecture Dossier V2”, is to define and design the MobiDataLab Transport Cloud architecture, a cloud-based prototype platform for sharing transport data using a federated approach. The architecture defined in this document allows to prototype a platform for searching, accessing and fusing multimodal mobility data in the cloud (WP4). This document balances long-term strategic issues with immediate requirements of the project’s current phase of technical development.

The first iteration of this document, i.e. the Deliverable D4.1, aimed to define the MobiDataLab Transport Cloud architecture, and was based (1) on the preliminary analysis of the use cases and datasets identified in the Deliverable D2.9 and (2) on the detailed analysis and evaluation of the relevant dimensions and state-of-the-art technologies and solutions (see Deliverable D2.6) that could be potentially leveraged to realise the Transport Cloud.

The second iteration, i.e., the Deliverable D4.2, provides updates and additional details with respect to what has been defined in the prior iteration. More specifically, it specifies how the various non-functional, functional, and general requirements are implemented (Section 4), provide an overview on the technologies that have been selected for the implementation of the Transport Cloud Architecture (Section 5), and finally provide an update on the status of the most promising initiatives related to MobiDataLab (Section 6).

1.3. Intended audience & review process

The dissemination level of the Deliverable D4.2 is ‘public’ (PU). CNR as Task 4.1 leader is responsible for it with the contribution of HERE, AKKODIS, URV, and HOVE. An external review is conducted by members of the Advisory Board.

Funded by the European Union

1.4. Structure of the deliverable and its relation with other work packages/deliverables

The document is organised as follows: Section 2 presents the main user roles to be supported by the platform, along with their main operations within the platform. Section 3 analyses the main information flows that will be executed over the platform, proceeds to the presentation of the architectural design of MobiDataLab (placing the components participating in the flows to distinct and clearly defined modules) and concludes with a summary of the technologies to be used for implementing the core components of the software stack. Section 4 describes the non-functional requirements from the software stack, the functional requirements, and the general requirements. Section 5 presents the technologies that have been selected for the implementation of the MobiDataLab components. Section 6 provides an update on the status of the promising initiatives related to MobiDataLab. Finally, Section 7 draws the conclusions.

The Deliverable D4.2 is related to all the other Deliverables of the WP4, and to the “Use cases definition V1” (Deliverable D2.9).

2. MobiDataLab Platform Overview and Main Actors

MobiDataLab targets technology challenges aimed to propose to mobility stakeholders a replicable methodology and sustainable tools that foster the development of a data sharing culture in Europe and beyond. Facing such challenges requires the use of techniques that deal with data fusion, privacy and anonymisation, geographical and semantic enrichment, harmonisation/standardisation, and data processing in general. These will be indeed required to improve the quality, accessibility and usability of mobility data, but also to allow each mobility data provider to share securely and safely their data.

The Transport Cloud will demonstrate a cloud-based prototype platform for sharing transport data and that is accessible to interested mobility actors. It is technically designed according to federated cloud principles. The MobiDataLab platform will showcase how to facilitate access to mobility data in an open, interoperable, and privacy preserving way through the development of open and accessible tools. The Transport Cloud is primarily designed to demonstrate and offer solutions to reduce and, in some cases, remove current technical limitations identified as barriers to data sharing and reuse.

We identify four (4) main actors who are expected to interact through the MobiDataLab platform: Administrator, Developer, Data Consumer and Data/Service Provider. The core activities for each actor are summarised as follows:

- **Administrator**: User management; Access management; Applications and Components configuration;
- **Developer**: Transport Cloud component deployment integration;
- **Data Consumer**: any entity that is interested to use the data and services available within the Transport Cloud. Some examples that can be observed or inferred from the use cases provided in the Deliverable D2.9 are:
    - Data scientist, researcher, domain expert;
    - Transport customers;
    - Services external to the Transport Cloud and that use its data and services.
- **Data/Service Provider**: any entity that provides, either passively or actively, data or services to the Transport Cloud. Some examples that can be observed or inferred from the use cases in the Deliverable D2.9 are:
    - Trip planners (e.g., Navitia, HERE);
    - MobiDataLab stakeholders, e.g., transport operators or public institutions that actively share their data and services for the good of the MobiDataLab project;
    - Open data/services providers (e.g., OpenStreetMap).

Funded by the European Union

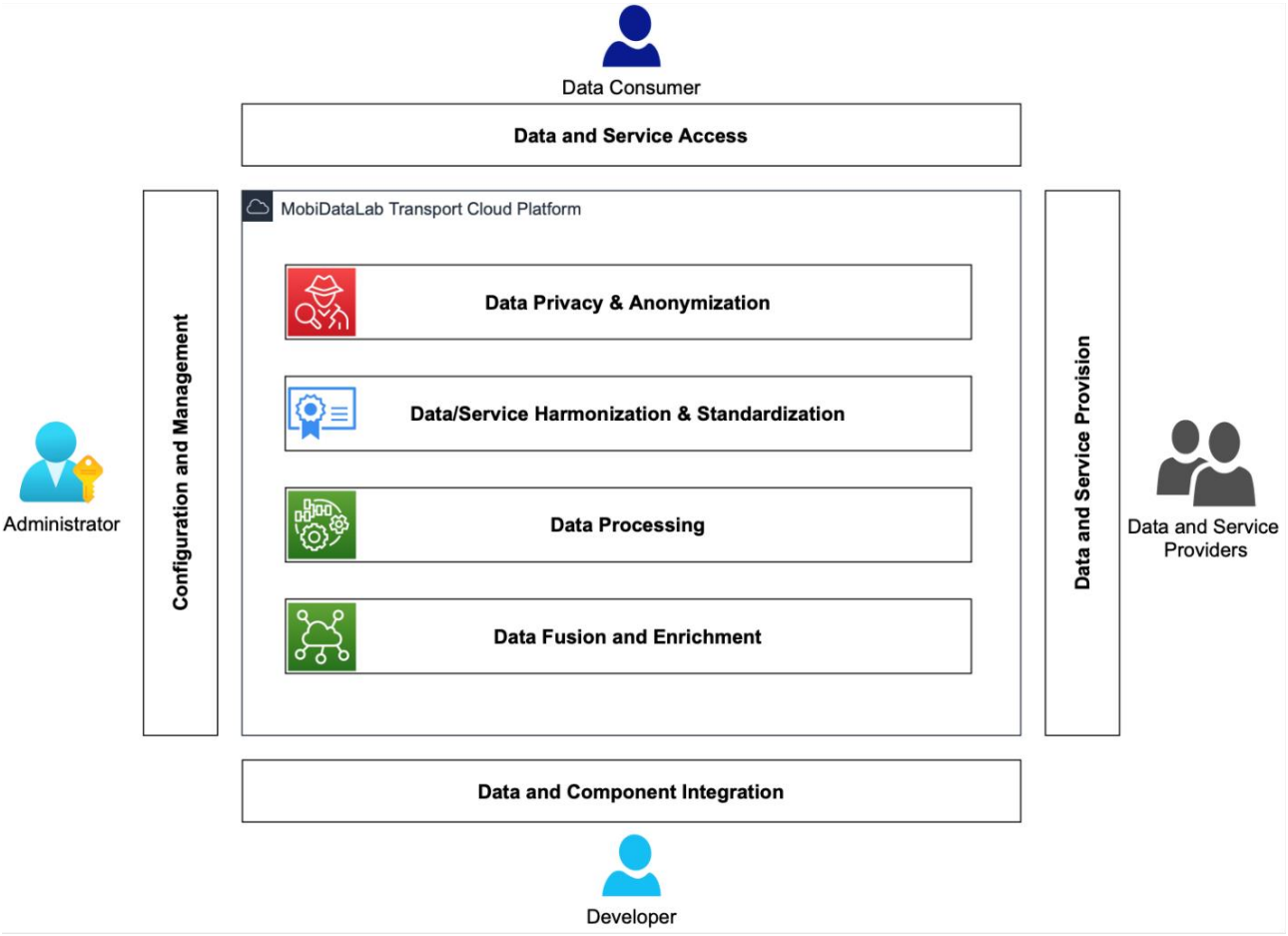

Figure 1: Logical Organisation of the MobiDataLab Transport Cloud platform

Figure 1 illustrates the logical organisation of the Transport Cloud platform. The platform comprises components that implement a set of functionalities required by the use cases and that will be presented in more detail in Section 4.

The components depicted in the figure will be exposed via clearly defined and thoroughly documented APIs and deployed on, as well as integrated in, the Transport Cloud platform, which in turn will serve the MobiDataLab use cases.

3. Transport Cloud Architecture

Based on the requirements reported in Section 4, a detailed Transport Cloud architecture is depicted in Figure 2. The Figure provides a graphical representation of the main components and the connections between them.

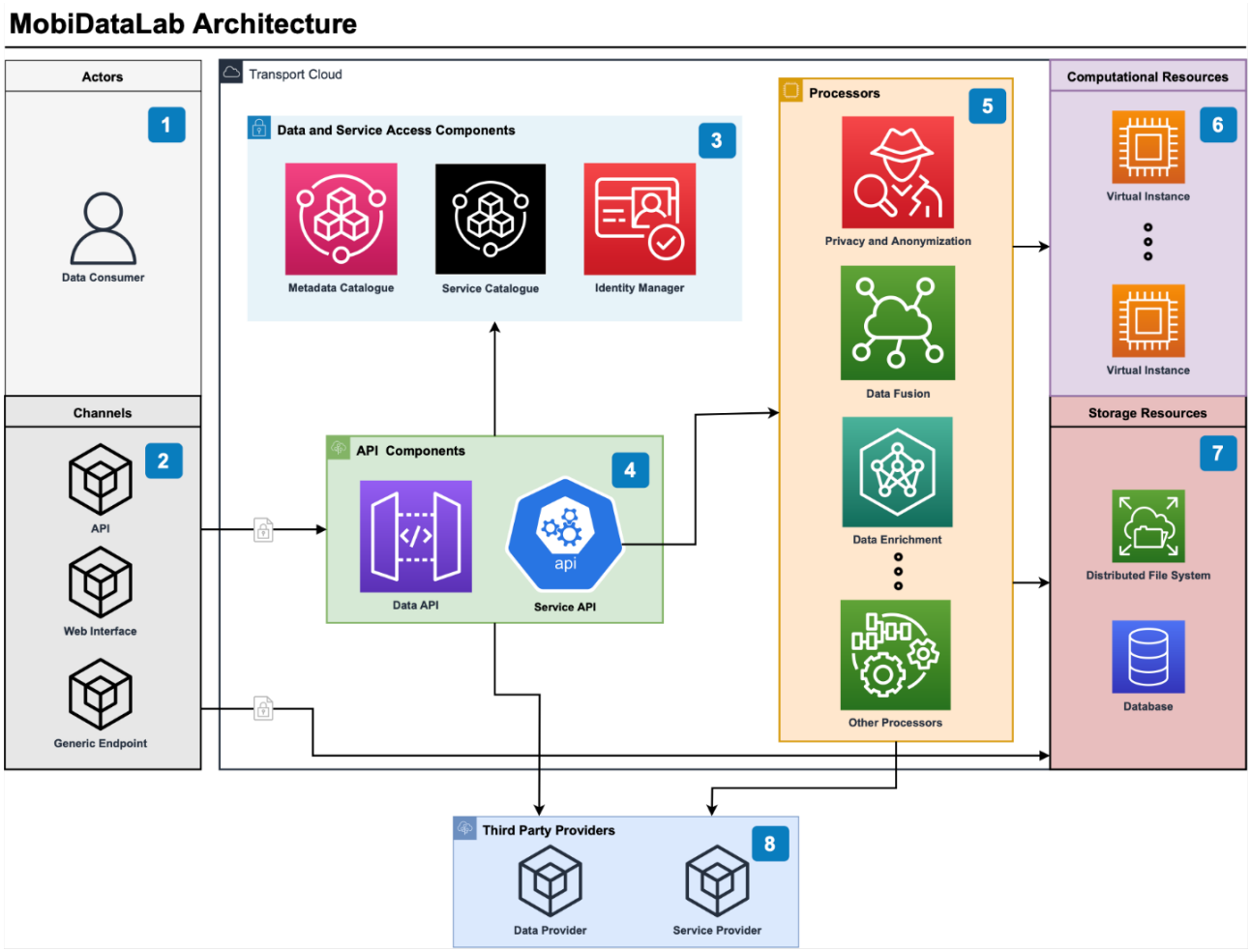

Figure 2: MobiDataLab Transport Cloud architectural design

The actors interacting with the platform that were introduced in Section 2 are present in Figure 2 under the general umbrella of “Data Consumer” (box 1) and “Third-party Providers” (box 8).

The Data Consumer can interact with the platform through several Transport Cloud channels (see also box 2 in Figure 2), i.e., (1) **API endpoints** (mainly dedicated to REST API services), (2) **web interface endpoints**, i.e., dedicated to services which need interaction with the end user (for example scenarios involving data analysis and visualisation tasks), and (3) **generic endpoints** - for instance, a SPARQL endpoint may enable a Data Consumer to access some knowledge base via RDF queries.

The various internal components associated with **Data and Service Access** (box 3), **APIs** (box 4), and **Data Processors** (box 5) lean on the usage of **Computational** (box 6) and **Storage** (box 7) **Resources**. By computational resources we refer to resources providing computation capabilities, i.e., virtual instances provided by the chosen cloud provider and intended to support the execution of the services deployed and running within the Transport Cloud platform. By storage resources, on the other hand, we refer to resources that enable the storage of information within the Transport Cloud such as distributed file system solutions and generic database solutions (e.g., PostgreSQL with PostGIS extension, SPARQL engines, and so on).

The subsections that follow will introduce the purpose and scope of each component of the architecture. More specifically, Section 3.2.1 introduces the Data Service and Access Components (box 3 in Figure 2). Section 3.2.2 introduces the API Components (box 4 in Figure 2). Finally, Section 3.2.3 introduces the notion of processors (box 5 in Figure 2), along with some specific processor specialisations that are fundamental to the goals of the MobiDataLab project.

3.1. Information Flow

The actors interacting with the Transport Cloud (previously introduced in section 2) will exchange information in ways defined by specific workflows, as derived by the general use case requirements.

The platform's information entry point is always represented by the third-party providers (box 8), who are responsible for the provision of information in the form of datasets and services. The access mechanism to these information sources should be identified and implemented according to the operations to be performed and on the types of data that need to be accessed.

Information retrieved from third-party providers can either be imported within the Transport Cloud, thus requiring appropriate storage solutions (e.g., relational databases, spatial databases, knowledge graph databases), or directly accessed by the Transport Cloud through the use of specific data and service endpoints being exposed by the providers. We report that the latter type of access is conducted by the Transport Cloud by means of the data and service APIs components. The Transport Cloud might also employ proper caching mechanisms to improve the efficiency of such type of accesses.

Data accessed through the Transport Cloud may need some form of processing such that the resulting data can be fruitfully exploited to implement the functionalities provided by the Transport Cloud and to satisfy the Data Consumers – a few relevant examples are fusion, enrichment, anonymisation, and format translation and standardisation. To this end, specific processors are identified and implemented, providing such functionalities to the Transport Cloud.

The Transport Cloud finally provides access to the several data sources and services the data consumers may need by exposing several types of channels implemented through specific API endpoints. More precisely, the data consumer interacts with the Transport Cloud first by authenticating themselves via the Identity manager.

Funded by the European Union

Once authenticated, the data consumer can proceed to submit their requests to the Transport Cloud, which then processes them by querying the metadata and service catalogues to find the appropriate data sources and services to satisfy the data consumer's needs.

3.2. Architectural Components

In this section we introduce the main architectural components onto which the Transport Cloud is built.

These components are essential to satisfy or realise the functional, non-functional, and generic requirements reported in Section 4.

3.2.1. Data and Service Access Components

This section presents the components (box 3 in Figure 2) which deal with identity management, and the discovery, exploration, and research of data and services available within the Transport Cloud.

Serving mobility data faces challenges that could be described as follows:

- Data volume: Consumption of mobility data (e.g., through mobile devices) incurs permanent activity between the application and the information system and that accumulation entails a high volume of queries to be processed;
- Data accuracy: real time activity such as availability (bikes, scooters, car park, etc.) must be trustworthy for the user to be sure to find on site the promoted asset when necessary;
- Data ownership: mobility data is a data stack valorisation layer on top of layers from different sources which access may not be totally free of charge or free to be publicly made available to everyone.

We do distinguish between:

- Metadata Access Components: Metadata Catalogues as described in Section 3.2.1.1.
- Data Access components: services providing access to the data. The access is realised open, or via authentication/authorisation. The data exchange is via proprietary or via interoperable and open interfaces (such as OGC specified service interfaces). The data formats themselves should be well known and ideally open, in a way that the data specification is available and usable for data consumers. These components are described in Section 3.2.1.2.
- Service Access components: services that provide relevant functionality for mobility challenges, e.g., journey planners, ticketing services, mobility data visualization, gazetteer services, routing services or further location-based services. These components are described in Section 3.2.1.2.

3.2.1.1. Metadata Catalogue

A metadata catalogue listing the content made available by the MobiDataLab project, such as datasets or metadata, will be presented to the Data Consumer user. This catalogue will be presented regardless of data volumes and ownership. The MobiDataLab client applications will take advantage of the centralisation brought by the catalogue to guide the user to gathered data content.

The metadata catalogue provides access to the mobility datasets via a web portal, and thus not only via an API. It is built on widespread solutions in the transport sector (see also Section 5) and expected to be used by several stakeholders. The management of spatially referenced resources (i.e., geospatial metadata) may be needed, as mobility data is often location-based. This support may be added or plugged to the standard catalogue via APIs or extensions.

The diversity of data types present in the mobility domain exposes the Transport Cloud Platform to technical challenges. These challenges must be addressed in order to offer qualitative services to data consumers, and to comply with the requirements imposed by the data sources.

To achieve the goal of bringing a wide variety of metadata and datasets, we incorporate two complementary technologies in the Transport Cloud:

- GeoNetwork
- CKAN

Using together these two data services allows to:

- Support a wide range of data format.
- Fuse the Data collection functionalities provided by the two services.

The functionalities of these services have been enhanced through:

- The interconnectivity of geospatial components with geospatial tools through dedicated endpoints, for instance a CSW endpoint, available and exposed in one or both services to the Transport Cloud users.
- The possibility of querying geospatial content with dedicated query language, for instance via PostGIS which is an extension of PostgreSQL dealing with spatial data.

3.2.1.2. Data Access and Services, Service Catalogue

Access to data and services is a critical part of the whole infrastructure of the Transport Cloud. For the data access, many data providers already built their data offers using standard formats and interfaces to make their data available.

For instance, Within the context of INSPIRE, a European initiative aimed to create a spatial data infrastructure for the purposes of EU environmental policies and policies or activities which may have an impact on the environment, OGC standardised interfaces for data and services are being used.

In some cases, “proprietary” data formats – developed for feature richness, interaction with internal workflows, or for other “historical” reasons – are used. Examples of these are findable in the HERE platform (https://platform.here.com/, https://developer.here.com/documentation) or the ESRI ArcGIS hub (https://hub.arcgis.com/). Further examples for such type of data and services are provided in the “D2.6 Report on enabling technologies for Transport Cloud”.

Typically, the services ecosystem is more heterogeneous and less standardised than the data ecosystem. However, in a world that is moving towards platform and service-oriented architectures, there is further need to make the services findable, accessible, interoperable, as well as reusable for several use cases. With the availability of interface descriptions (e.g., OpenAPI), services are becoming more transparent and a potential to provide metadata for them has grown. Nevertheless, the standardisation of the interfaces needs to be addressed within this part to show the possibilities of interoperability and potential for adjustments and evolution of the services.

These challenges will be analysed and addressed to extend availability and interoperability of data and services.

3.2.1.3. Identity Management

The MobiDataLab Transport Cloud will ensure Identity and Access Management (IAM), which may include identity provisioning, authentication, and authorisation. In the context of mobility data sharing, these mechanisms can be favourably integrated into an API gateway solution (see also the elicited technologies in Section 5). For clarification purposes, these different concepts are defined below:

- **Identity Provisioning** is the process of creating, updating, retrieving and deleting users’ digital identities. Digital identity is carrying information such as username, password, email, address, phone number, roles, groups, etc. about the registered users;
- **Authentication** is the process of identifying an individual, entity, or website based, for example, on a username and password, a client identifier and client secret or biometric data (fingerprint, iris scan, face recognition, etc.). In security systems, authentication is distinct from authorisation, which is the process of giving individuals access to system objects based on their identity. Authentication merely ensures that the individual is who they claim to be, but says nothing about the access rights of such individual;
- **Authorisation** is the process of granting or denying access to a resource. Most computer security systems are based on a two-step process. The first stage is authentication, which ensures that a user is who they claim to be. The second stage is authorisation, which gives to the user access to various resources based on their identity;
- **SSO** (Single Sign-On) is an authentication process in a client/server relationship where the user, or client, can authenticate once (with only one user/password pair for example) and have access to more than one application, or access to a number of resources within an enterprise or the whole web. Single Sign-On takes away the need for the user to enter further authentications when switching from one application to another.

Funded by the European Union

3.2.2. API Components

The API components (box 4 of Figure 2) consists of two parts: the Data API and the Service API. Within the catalogues described in Sections 3.2.1.1 and 3.2.1.2, there is a list of available data and services including their metadata. To access the services, a direct connection is needed: the API components therefore take the metadata from the catalogues, including the references connecting to the services and accessing the data.

The first step is to make data and services available by following a standardised and open interfaces approach. Then, the challenges of integrating non-standardised interfaces and data must be analysed. The different options for integrating proprietary data and service APIs will be explored based on the data and services provided by the project's stakeholders (i.e., the MobiDataLab reference group).

Globally, the following family of APIs have been identified:

- **metadata catalogue**: lists metadata and brings filtering for sharper search.
- **service catalogue**: lists the set of services hosted by the platform
- **basic data access** API: provides access to data that does not need « processing » such as public datasets or Navitia input data.
- **advanced data and service access** API: provides access to dataset “post processing ", such as the dataset requiring trajectories logs anonymisation generated by Navitia.

Among the set of APIs, those that are particularly crucial are:

- “Transport and Accessibility APIs” to find, browse, and explore datasets as described in Section 4.2.6(FR-F.01).
- “Environmental datasets APIs” for metadata catalogue available in the Transport Cloud, to provide functionalities that allow to find, browse, and explore datasets as described in Section 4.2.7(FR-G.01).

Accordingly, it is important to add all the APIs related to:

- Infrastructure:
    - “Authentication APIs” related with the identity management, aiming to identify data consumer from the credentials and will therefore apply user profile specific feature but also allow rate management to prevent the Transport deny of service by query flooding.
    - “Session management APIs”, in charge with the data consumer experience personalisation and activity contextualisation anonymously identified with a persistent UserID.
- Historization:
    - “User dataset historization APIs” for comparing the newly incoming query with previously stored query before, eventually, sending back the previously stored matching data as expected from the use cases, i.e., the functional requirements:

- Dataset combination and Enrichment:
    - “User query result storage recovery APIs” aimed to bring consistent result to the client/user of the Transport Cloud Platform user/client. See Sections 4.2.6(FR-F.02) and 4.2.7(FR-G.02).

3.2.3. Processors

We define a processor within the Transport Cloud to be a component that models some function, that operates on some input data according to some specific logic in order to produce a final output. In Figure 2, processors are showed in box 5. Such a definition can be used to instantiate the notion of a processor in several different ways. In the context of mobility data sharing, data processors may be needed to perform, e.g., semantic enrichment based on common vocabularies, geographical enrichment based on common geometries, data format translation, data fusion, data anonymisation, injection of license specification, and any other data processing tasks that are relevant to the goals of the project.

3.2.3.1. Data Enrichment Processors

Positioned at the centre of a maze of information flows, the MobiDataLab project intends to fulfil two missions:

- To be a standard and universal entry point to various actors in the mobility industry;
- To provide users with mobility data enriched with on-demand elements from third-party partners.

Enhancing mobility data means adding optional information to the API responses including e.g., points of interest around the end-user, their accessibility to wheelchairs and strollers, the availability of bike sharing services, battery chargers, the weather conditions, etc.

This on-demand enrichment is achieved thanks to:

- APIs entry point: the user will specify here the geographical focus point for which he/she wants to obtain mobility-related information;
- The processing back-office unit: it will consult the content of the data supplied by the data providers, assemble and render the set of features that matches the location; then it will produce the response in the requested format and allow the API to respond to the user.

3.2.3.2. Privacy, anonymisation

The privacy and anonymisation components in the MobiDataLab Transport Cloud are responsible for assessing the privacy risks in the information flow through the computation of privacy metrics on

transport data (such as the unicity of trajectories, see Deliverable D2.3: "State of the art on Mobility and Transport data protection technologies" for more details), and the execution of anonymisation procedures, both for real-time data and for data stored in the Transport Cloud.

Data anonymisation irreversibly transforms data in a “privacy preserving” way. The outcome of the data anonymisation process is data that can be used by other users, but without disclosing any personal or private information included in the original data. An alternative to anonymisation is to generate synthetic datasets based on original historical data. Synthetic data should maintain the statistical measures of the original dataset.

Data anonymisation is performed at the storage stage; after that, any query on the data (search, retrieval, calculations) are transparent, even though they may result in approximate results. Anonymisation can also occur in real-time, but this comes at a price: not knowing the distribution of data to protect will make it more difficult to achieve acceptable levels of privacy and utility.

Anonymisation can also occur in real-time, but this comes at a price: not knowing the distribution of data to protect will make it more difficult to achieve acceptable levels of privacy and utility.

Three modes of anonymisation mechanisms are proposed:

- Real-time data anonymisation often relies on the distortion or generalisation of user positions, or on the partition of real-time user trajectories;
- Historical or aggregated data anonymisation typically involves the transformation of data to achieve a privacy guarantee dictated by some privacy model, such as k-anonymity and its extensions or differential privacy.
- Synthetic mobility data generation based on original mobility patterns.

Funded by the European Union

4. Transport Cloud Requirements

This section summarises the various non-functional, functional, and generic requirements for the MobiDataLab Transport Cloud platform.

4.1. Non-Functional requirements

This section summarises the non-functional requirements for the MobiDataLab Transport Cloud platform.

4.1.1. Governance and Regulation aspects

The storage and processing of data by the MobiDataLab Transport Cloud must satisfy strict key requirements concerning the privacy and trust of European citizens and businesses.

The non-functional requirements presented below address these needs.

NFR-GOV.01	Data Sovereignty
Description	Data stored and processed by the MobiDataLab Transport Cloud shall be kept under the European Union authority, thus ensuring data sovereignty.
Objectives	• Data stored and processed by the MobiDataLab Transport Cloud shall be located within premises or data centres falling in the European Union territory. Data stored and processed by the MobiDataLab Transport Cloud shall be located within premises or data centres falling in the European Union territory.
Implementation	When selecting cloud providers, the MobiDataLab project considered only those that abide by the European laws and regulations, and which computational and storage resources are located within the European Union.

NFR-GOV.02	Data Anonymisation
Description	The MobiDataLab Transport Cloud must ensure that personal or sensitive data accessed or processed is adequately protected before sharing or releasing them.
Objectives	• The MobiDataLab Transport Cloud must implement anonymisation mechanisms that are employed when the

	platform needs to access or process data to execute some tasks requiring some form of anonymisation. Examples of types of data that are relevant to the MobiDataLab project are GPS locations, personal information such as e-mail, names, addresses, and so on; • Build confidence with end users over privacy; • Fulfil existing privacy regulations.
Implementation	Several anonymisation methods have been selected to be implemented among those depicted in D2.3: • Microaggregation • SwapLocation • SwapMob New techniques are also being developed: • Synthetic mobility data generation based on NLP • Privacy-preserving methods to compute aggregated data Utility and privacy metrics will also be provided.

4.1.2. Cloud Federation Aspects

The MobiDataLab Transport Cloud will strive to represent a concrete example of viable cloud federation by satisfying the non-functional requirements introduced below.

NFR-CFA.01	**Transport Cloud agnosticism**
Description	A major requirement which needs to be satisfied to validate the MobiDataLab project calls on the Transport Cloud to be agnostic with respect to any suitable cloud provider that may be used to support its implementation.
Objectives	• The MobiDataLab transport cloud's design must ensure that the Transport Cloud does not require nor depend on any specific cloud editor. As a consequence, the Transport Cloud can be deployed on any suitable cloud technology vendor infrastructure.
Implementation	The design of the transport cloud architecture has been conducted to avoid any dependency to specific cloud providers.

NFR-CFA.02	**Use of open source, standard technologies**

Description	Serialisation and standardisation are key tenets that shall underpin the MobiDataLab transport cloud's design. These have clear benefits in terms of costs, wide availability, and open documentation, thus paving the way to technology and knowledge transfer.
Objectives	• Select and leverage well-established tools, technologies, and processes that align with the aforementioned tenets; • Ensure compatibility with the external world.
Implementation	The technologies that have been selected for the design and the implementation of the transport cloud according to the principles detailed in the requirement description.

NFR-CFA.03	**Cloud implementation strategy**
Description	Cloud providers differ from one another in the number and types of services they provide, as well as in the way said services are set up within each cloud platform. The MobiDataLab project aims to select a set of services common to all main cloud providers. Such selection shall comprise standard information technology tools required to build the Transport Cloud infrastructure, regardless of whether the infrastructure will be hosted on premise, virtualised on some cloud platform, or some hybrid between these two approaches.
Objectives	• Determine the set of services that are common among the major cloud providers and that can be used to implement the functionalities the Transport Cloud shall provide; • Guarantee that the Transport Cloud's functionalities can be implemented regardless of the approach chosen to implement the underlying infrastructure.
Implementation	The MobiDataLab project has conducted activities that focused on finding out the set of services common among the major cloud providers, and that can be used to implement the Transport Cloud functionalities.

4.1.3. Data Management Aspects

NFR-DM.01	**Data distribution**
Description	Data distribution is a fundamental principle upon which the MobiDataLab project is built. The MobiDataLab project intends to offer two entry points for a multitude of variated data sources, either public or privately owned.

Funded by the European Union

Objectives	• Design the MobiDataLab Transport Cloud architecture so that it enables the vision highlighted in the description above.
Implementation	Since the Transport Cloud will gather data coming from several data providers, each having different localisations, interests, and expectations, the platform must be able to host a large variety of data according to the different consumers' needs. Accordingly, the activities that have been conducted to address this goal are: • an extensive work identifying data sources that match both technological and geographical consistency with the project's content target. • To ensure complementarity, many metadata and datasets were targeted in order to ensure post selection filtering and double eviction, thus avoiding the risk of missing relevant content. • Investigations to identify the added value for data types according to the metadata catalogue service hosted by the MobiDataLab project in order to take advantage of the most relevant content server in its category. • Regarding the data sources, high importance was given to public/open data providers, since private data sources come with commercial restrictions, as well as technical or regulatory constraints.

NFR-DM.02	**Data sources identification**
Description	The MobiDataLab project must identify data sources which provide the data needed by the Transport Cloud to satisfy the use cases presented in the Deliverable D2.9.
Objectives	• Identify suitable data sources to import open datasets, metadata, but also to create a data catalogue within the Transport Cloud. • Create rich content from multiple data sources.
Implementation	The following three-stage workflow has been conducted to satisfy the data source identification (prior to selecting an ideal base data to work with): • Work with local partners of the MobiDataLab project to understand their expectations and assess their involvement through the open data culture put in place by their administrations or by their partners (e.g. include the MobiDataLab Reference Group and the Advisory team in the analysis).

	• Assess the data quality and compatibility with current technologies to ensure exploitability and readability by all data consumers browsing the content hosted by the Transport Cloud Platform. • Import and categorize metadata or datasets from territorial portals that will be involved in the use cases discussed with the territorial administrations, or from the more generic use cases.

NFR-DM.03	**Data ownership**
Description	Using privately owned data, and possibly combining it with public open data, poses new challenges to the MobiDataLab Transport Cloud in terms of data ownership that needs to be settled among private data owners, the MobiDataLab project stakeholders, and the data consumers willing to use the Transport Cloud for the data and services that it provides.
Objectives	• Fulfil existing intellectual property regulations, striving to strike a good balance between the need to preserve intellectual property and the need to serve public institutions and the general public at large.
Implementation	Harvesting metadata or datasets from both public and private data sources creates an ownership dilemma in data exploitation and distribution. Indeed, private data owners are very dedicated to keep sovereignty over their immaterial properties. Consequently, data will be labelled and identified according to the following annotations: • Public data freely accessible to any data consumer. • Private data that requires a pre-authorisation before granting access to consultation. This pre-authorisation will remain under the data owner's regulation and will take the shape of (1) subscription from the client/user and (2) subscription granted to MobiDataLab to be shared with the Transport Cloud Platform visitors. • Data under the Transport Cloud jurisdiction that contains sensitive personal information and thus requires anonymisation before being put under public eyes.

4.2. Functional Requirements

In this section, we introduce the main functional requirements the MobiDataLab Transport Cloud is supposed to satisfy to cover the needs of the use cases introduced in the Deliverable D2.9.

4.2.1. Use Case A: Optimisation of Transport flow and ETA

FR-A.01	Transport flow monitoring
Description	In order to optimise, monitor, and manage commercial transport flow, it is crucial to have updates (either periodical or in real-time) from various data sources -- e.g., fleet status, weather, traffic, planned events, etc. -- to provide an overall picture of the commercial transport system and to trigger specific actions whenever needed, i.e., delays, arrival time sharing, and tour plan update.
Objectives	• The Transport Cloud must provide a **monitor processor** which interacts with the fleet IT system, from which the monitor receives a stream of information; • the ETA estimator, a service used by the Transport Cloud to update/optimise tour planning; • Other data sources such as weather, traffic information, planned events, and so on, which may further help to improve service quality provided by the fleet owner; • The Transport Cloud must provide **notification/planning functionalities** within the monitor that allow the Transport Cloud to achieve the goals specified in the description of this requirement.
Implementation	The Transport Cloud provides the capability to search for road network data to build a routing graph. Data on weather conditions, traffic, and planned events are available depending on the region and the individual data suppliers. • The available data sets can be combined within open source routing services (e.g., GraphHopper) and exposed via a web service interface or • via Geoinformation Systems, e.g., ArcGIS Network Analyst

FR-A.02	Tour reporting and planning
Description	Planned tours have to follow rest and break time regulations. The system should therefore be able to take these aspects into account when planning tours. Moreover, explicit reporting of already completed tours can be requested by the fleet owner to compare planned, initially estimated, and actual arrival times, thus allowing to identify potential problems.

	Differently from the activities described in requirement FR-A.01, the activities in this requirement are requested on demand and take advantage of information gathered by the monitor.
Objectives	• The Transport Cloud must provide a **REST-API component** allowing an external user (e.g., the fleet owner) to request on demand specific services to the Transport Cloud and receiving back the result; • The Transport Cloud must provide an **advanced dispatcher processor** that uses an ETA estimator and consider rest and break time regulations, to plan tours that are compliant with existing EU legislation.
Implementation	The Transport Cloud provides a list of service APIs that supports routing requests. On client side, the requests can be setup according to the break time regulations. In combination with places data (e.g., rest points), the tour planning can be adjusted to determine options for breaks and an efficient journey.

FR-A.03	**Data distribution**
Description	Data distribution is a fundamental principle upon which the MobiDataLab project is built. The MobiDataLab project intends to offer a single data entry point for a multitude of variegated data sources, either public or privately owned.
Objectives	• Design the MobiDataLab Transport Cloud architecture so that it enables the vision highlighted in the description above.
Implementation	Metadata catalogues are providing the options to search for data and access publicly available datasets.

4.2.2. Use Case B: Emission Reporting

FR-B.01	**Emission estimation and reporting**
Description	Reducing environmental impact is highly relevant for any form of mobility and transport. Concrete action for reducing the environmental footprint can only be taken in a systematic way if the impact is reported in a clear and transparent manner. Therefore, the need for predicting and reporting emissions is of great significance.
Objectives	• The Transport Cloud must provide an **emission processor** that is able to predict the emission footprint of a planned/already

	completed tour, based on the tour plan, telematics data collected by the involved vehicles (if available), and information concerning direct and indirect emissions; • The Transport Cloud must provide an exhaustive **emission report processor** that is able to take into account the emission estimation of a trip possibly composed of several tours (like it happens for the transportation of goods where several transport assets are used).
Implementation	Via the Transport Cloud, connections to platforms are offered, that provide routing capabilities. Emission reporting can be implemented in various details, depending on the specific user requirements and the available data: • Based on energy consumption models, a generic emission reporting can be build based on the route and vehicle • With more detailed vehicle data, connected/listed specialized route services from 3rd parties can be used to calculate standardized emission reports (depending on their availability)

4.2.3. Use Case C: Analytics & Learning

FR-C.01	**Data Access, Analytics and Learning**
Description	Being able to provide access, visualise, and finally analyse data is a crucial component in every data platform. To this end, this use case is providing a horizontal connection to all use cases, since analysis and learning methods can contribute to most of them.
Objectives	• The Transport Cloud must provide a **loader component** that must be able to load spatial data and could be able to load other types of non-spatial data (e.g., tabular, etc.) into the Transport Cloud; • The **metadata catalogue** available in the Transport Cloud must provide functionalities that allow to find, browse, and explore datasets that are of interest to the data consumer; • The **data API** must provide access to data sources containing datasets of interest to the data consumer and that do not require any preliminary data pre-processing (e.g., anonymisation) before being accessed; • The **service API** must provide access to data sources containing datasets of interest to the data consumer and that do require some preliminary data pre-processing (via a suitable processor) before being accessed.

Implementation	The MobiDataLab platform provides services, such as GeoNetwork, CKAN to find data and getting access to the data source. • Client components, such as QGIS or ArcGIS will provide a set of tools to access the data and run analysis algorithms on the data sets provided via the Transport Cloud. Furthermore, the client tools provide options to visualise data and present the data for decision makers • Tools like Python SDKs, R, and external data analytic tools (such as kepler.gl) will provide further options for specific data analytics or more sophisticated visualisations (e.g., 3D)

4.2.4. Use Case D: Re-use of transport data for journey planners / digital services

FR-D.01	**Multi-modal Journey Planning**
Description	Multi-modal journey planning capability is a feature of interest for many digital service providers. However, dealing with raw transport datasets, using different formats and combining them could be particularly difficult and complicated. To this end, a common solution for journey planning would therefore foster the use of transport data and simplify its usage.
Objectives	• The Transport Cloud must provide a **loader component** that allows to load data into the Transport Cloud from public transport operators/authorities. The loader should also allow to load data concerning POIs or from other transport modes; • The Transport Cloud must provide a data converter processor that converts the aforementioned data into a standard data format. This processor must also integrate different data sources. Finally, the processor should be able to process and enrich datasets. The processor must finally provide the end result of its activities available to other entities; • The Transport Cloud must provide a **journey planning processor** using integrated datasets. The planner might be able to use third party external calculators. Finally, the planner should be able to return journey statistics and provide journeys through a standardised API.
Implementation	The MobiDataLab platform will achieve these objectives by using the following tools (provided by Hove): • The Navitia open-source journey planning processor (available through Navitia API)

	• Data Hub, which is a mobility data loader and processor tool Navitia is able to use other journey planning processors for computing multi-modal journeys. In particular, it could be integrated with the Here road planning processor. Data Hub is able to integrate mobility datasets and POIs from different sources, combine them, enrich them, and output them using standard formats (GTFS, SIRI). Once processed, these datasets can be integrated in journey planning processors such as Navitia. At the moment, Data Hub handles only public transport data sets and OSM POIs. To integrate other datasets such as road traffic data or weather data, other loader components should be used or implemented, e.g., the loader components that may be proposed by Here for road traffic.

4.2.5. Use Case E: Mobility as a Service (MaaS)

FR-E.01	**Mobility As A Service**
Description	Mobility is a continuously evolving service, including a variety of heterogeneous transport solutions (bus, train, car sharing, bikes, etc) provided by different providers (public and private). Thus, it is crucial to provide users with an end-to-end solution encompassing journey planning and ticketing. Within the MobiDataLab project, we aim to provide access to raw datasets on the one hand, and on the other hand to provide a multi-modal journey planning service.
Objectives	• The Transport Cloud must provide a **loader component** for loading data into the Transport Cloud from raw transport, journey planning, and MaaS datasets. This functionality shows the need of dedicated storage solutions to be deployed within the Transport Cloud • The Transport Cloud must provide access to datasets imported from MaaS operators - this can be achieved by means of the **metadata catalogue** and the **data/service APIs.** • The Transport Cloud should provide a **data transformer processor** that is able to integrate booking and payment data from the MaaS operators and could integrate journey data as well. • The Transport Cloud must provide a **journey planning processor**, eventually using third party external calculators,

	implementing the service and providing the response in a standard format through a MaaS paradigm.
Implementation	The loader component and the data transformer processor for journey planning will be the same as for the use case D: “multi-modal journey planning”. In addition, more loader components may be proposed to integrate MaaS specific datasets, e.g., fare datasets (could be integrated through Hove Data Hub or a specific loader component) Journey processing data will be made available through the platform to any user with the correct access rights, along with datasets imported from MaaS operators, thus allowing third parties to use these datasets for statistical and data mining purposes.

4.2.6. Use Case F: Geodata Sharing applied to Transport: OpenStreetMap for Inclusive transport

FR-F.01	**Dataset discovery**
Description	The data consumer uses the Transport Cloud to find, browse, and explore transport and accessibility datasets that can suit their needs.
Objectives	• The metadata catalogue available in the Transport Cloud must provide functionalities that allow to find, browse, and explore datasets that are of interest to the data consumer; • The **data API** must provide access to data sources containing datasets of interest to the data consumer and that do not require any preliminary data pre-processing (e.g., anonymisation) before being accessed; • The **service API** must provide access to data sources containing datasets of interest to the data consumer and that do require some preliminary data pre-processing (via a suitable processor) before being accessed.
Implementation	Offering a wide variety of datasets would not make any sense if the Transport Cloud data consumer was not able to have knowledge of the list of datasets available via the platform. Consequently, an API is devoted to browse the metadata selection on the metadata servers. Once the user/client has identified the metadata of interest, the data consumption will be redirected to the data entry point described by the metadata content.

	• The raw data consumption, prior to enrichment, will be accessible via API for reaching public data that does not need to go through anonymisation techniques. • Data needing cautious handling due to sensitive content that may need to go through the anonymisation processor will be accessed by redirecting the data consumer to the data source. • The accessibility distinction within the Transport Cloud platform will come from (1) datasets labelled or flagged from the dataset catalogue as accessibility capable or accessibility oriented and (2) a journey planner that includes accessibility in the stop description on the city where the journey takes place. • Section 4.3 provides further details on this subject, more specifically in the FR-GENERIC.05 / Basic Data Access API requirement and the FR-GENERIC.06 / Advanced Data and Service Access API requirement.

FR-F.02	**Dataset combination and enrichment**
Description	The data consumer wants to use the Transport Cloud to combine and enrich (either geospatially or semantically) the datasets identified via the functionalities described in FR-F.01. The result of these operations consists in one or more enriched and consolidated datasets.
Objectives	• The Transport Cloud must provide a **dataset joiner** whose purpose is to perform the fusion of the datasets that are of interest to the data consumer. This joiner should be able to combine mobility data (e.g., provided by public transport authorities) with OpenStreetMap accessibility data following the OSM common exchange format including "tags". Such a combination should rely on the identification of a common location (so-called "node" in OSM terminology). • The Transport Cloud must provide a **geospatial enrichment processor** whose purpose is to perform the geospatial enrichment of datasets; • The Transport Cloud must provide **storage mechanisms** to store the enriched and consolidated datasets, so that the data consumer can retrieve them later.
Implementation	The MobiDataLab is a meeting point gathering different tools and technologies format intended to be complementary and/or supplementary. • The Transport Cloud Project brings together processors as well as journey planners which allow the data consumer to fetch, combine and enrich the datasets. These operations are provided by specific processors implemented within the Transport Cloud.

	• The OSM and PBF formats are the geographical support integrated to the Transport Cloud consolidation terrain. • User access is session oriented and uses a unique identifier per client/user. This allows to store and provide historical information about the submitted queries.

FR-F.03	Data analysis
Description	The data consumer wants to retrieve and then analyse with its preferred analysis tool, the enriched and consolidated dataset(s) produced via the functionalities described in FR-F.02.
Objectives	• The Transport Cloud must provide **storage mechanisms** to store the enriched and consolidated datasets, so that the data consumer can retrieve them later via the **metadata catalogue** and the **data/service APIs**.
Implementation	Based on the authentication principle to grant the use of the Transport Cloud Platform APIs, it is assumed that the data consumer will be subject to request historisation, which allows to keep track of the consumer's previous queries. This in turn allows the Transport Cloud Platform to store client/user requests, and to resupply these answers on demand. This functionality is also useful for the MobiDataLab Project to optimise resources and to lower the response time of the implemented services.

4.2.7. Use Case G: Geodata Sharing applied to Transport: Environmental Data for Sustainable Transport

FR-G.01	Dataset discovery
Description	The data consumer uses the Transport Cloud to find, browse, and explore transport and environmental datasets that can suit their needs.
Objectives	• The **metadata catalogue** available in the Transport Cloud must provide functionalities that allow to find, browse, and explore datasets that are of interest to the data consumer; • The **data API** must provide access to data sources containing datasets of interest to the data consumer and that do not require any preliminary data pre-processing (e.g., anonymisation) before being accessed;

	• The **service API** must provide access to data sources containing datasets of interest to the data consumer and that do require some preliminary data pre-processing (via a suitable processor) before being accessed.
Implementation	Offering a wide variety of datasets would not make any sense if the Transport Cloud data consumer was not able to have knowledge of the list of datasets available via the platform. Consequently, an API is devoted to browse the metadata selection on the metadata servers. Once the user/client has identified the metadata of interest, the data consumption will be redirected to the data entrypoint described by the metadata content. • The raw data consumption, prior to enrichment, will be accessible via API for reaching public data that does not need to go through anonymisation techniques. • Data needing cautious handling due to sensitive content that may need to go through the anonymisation processor will be accessed by redirecting the data consumer to the data source. • The accessibility distinction within the Transport Cloud platform will come from (1) datasets labelled or flagged from the dataset catalogue as accessibility capable or accessibility oriented and (2) a journey planner that includes accessibility in the stop description on the city where the journey takes place. • Section 4.3 provides further details on this subject, more specifically in the FR-GENERIC.05 (Basic Data Access API) and the FR-GENERIC.06 (Advanced Data and Service Access API) requirements.

FR-G.02	**Dataset combination and enrichment**
Description	The data consumer wants to use the Transport Cloud to combine and enrich, either geospatially or semantically, the datasets identified via the functionalities described in FR-G.01. The result of these operations consists in one or more enriched and consolidated datasets.
Objectives	• The Transport Cloud must provide a **dataset joiner** whose purpose is to perform the fusion of the datasets that are of interest to the data consumer. This joiner should be able to combine mobility data (e.g., provided by public transport authorities) with local environmental data following the Geographical Information Systems formats and exchange standards (OGC, INSPIRE, etc). Such a combination should rely on the identification of a common location (so-called "geometry" in GIS terminology).

	• The Transport Cloud must provide a **geospatial enrichment processor** whose purpose is to perform the geospatial enrichment of datasets. • The Transport Cloud must provide **storage mechanisms** to store the enriched and consolidated datasets, so that the data consumer can retrieve them later.
Implementation	The MobiDataLab is a meeting point gathering different tools and technologies format intended to be complementary and/or supplementary. • The Transport Cloud Project brings together processors as well as journey planners which allow the data consumer to fetch, combine and enrich the datasets. These operations are provided by specific processors implemented within the Transport Cloud. • The OSM and PBF formats are the geographical support integrated to the Transport Cloud consolidation terrain. • User access is session oriented and uses an unique identifier per client/user. This allows to store and provide historical information about the submitted queries. • The OGC implementation is integrated in the MobiDataLab project by the metadata service connectivity opened to the QGIS client. This allows geographical data representation of selected datasets as well as query refinement and filtering done by the QGIS visualizer. • PostgreSQL database server combined with its PostGIS extension are dedicated to geospatial data modelling. This allows SQL queries to browse structured data tables for any GIS capable browser or API client that interfaces with the Transport Cloud Platform and visualise graphically the query output based on the database content.

FR-G.03	**Data analysis**
Description	The data consumer wants to retrieve and then analyse with their preferred analysis tool the enriched and consolidated dataset(s) produced via the functionalities described in FR-G.02.
Objectives	• The Transport Cloud must provide **storage mechanisms** to store the enriched and consolidated datasets, so that the data consumer can retrieve them later via the **metadata catalogue** and the **data/service APIs**.
Implementation	Based on the authentication principle to grant the use of the Transport Cloud Platform APIs, it is assumed that the data consumer will be subject

	to request historisation, which allows to keep track of the consumer's previous queries. This in turn allows the Transport Cloud Platform to store client/user requests, and to resupply these answers on demand. This functionality is also a means for the MobiDataLab Project to optimise resources as well as response time. Furthermore, the data analysis application will combine the Python language portability with the Python Geopandas library for analysis tools development.

4.2.8. Use Case H: Transport Data Sharing within the Linked Open Data vision

FR-H.01	**Dataset provision and discovery**
Description	Some of the actors involved in this use case may need to either provide datasets to the transport cloud, or find, browse, and explore datasets or data sources that can be used to enrich or complete their own datasets.
Objectives	• The Transport Cloud must provide a **loader component** that allow data providers to load the datasets they need to provide to the Transport Cloud; • The Transport Cloud must provide **storage mechanisms** that allow data providers to store the datasets they need to provide on the Transport Cloud; • The Transport Cloud must be able to query data gathered in semantic databases via e.g., SPARQL queries.
Implementation	For what concerns the loader component, we refer the reader to the generic functional requirement FR-GENERIC.02 in Section 4.3. Storage resources and mechanisms can be provided using the infrastructure of the cloud provider used by the transport cloud. To this end, we refer the reader to the generic functional requirement FR-GENERIC.01 in Section 4.3. Finally, the transport cloud can implement a type of processor that executes a triple store able to query semantic databases, e.g., GraphDB.

FR-H.02	Data combination and enrichment
Description	The tourism service provider wants to combine and enrich geospatially or semantically, the datasets identified via the functionalities described in FR-H.01. The result of these operations consists in one or more enriched and consolidated datasets.
Objectives	• The Transport Cloud must provide a **semantic combination processor** whose purpose is to perform the fusion of the RDF datasets that are of interest to the data consumer. This processor should be able to combine data published according to Linked Open data principles. Such a combination should rely on a common vocabulary (so-called ontology); • The Transport Cloud must provide a **semantic enrichment processor** which purpose is to perform the semantic enrichment of datasets.
Implementation	Both objectives are targeted by the semantic enrichment processor, which is part of the task 4.4.

FR-H.03	Tourism analytics
Description	The tourism service provider wants to perform tourism analytics on the dataset(s) produced via the functionalities described in FR-H.02. Such analytics must be performed within the Transport Cloud.
Objectives	• The Transport Cloud shall provide a **processor** whose purpose is to perform **tourism analytics** according to specifications provided by the tourism service provider; • If required, the Transport Cloud may need to provide **storage mechanisms** to store the results within the Transport Cloud, thus allowing the tourism service provider to retrieve them at a later stage.
Implementation	The processor in charge of performing tourism analytics will host some selected triplestore capable of querying RDF datasets, thus enabling the possibility to conduct analyses on said datasets. Indeed, suitable triplestores allow to import RDF datasets, query said datasets according to the SPARQL query language, and possibly use Linked Open Data sources (via the notion of federated query) to further augment the analyses being conducted.

Funded by the European Union

4.3. General Functional Requirements

FR-GENERIC.01	Storage capability
Description	Actors interacting with the Transport Cloud, or components within the Transport Cloud, may require storing data within the platform to support their activities.
Objectives	• The Transport Cloud must provide **persistent storage mechanisms** to store the enriched and consolidated datasets, so that the data consumer can retrieve them later. Storage should be **scalable** and **fault tolerant** to the data access. • These storage mechanisms may vary according to the actors' or components' needs and the type of data being considered. To this end, we refer the reader to Sections 6.1 and 6.2.
Implementation	The Transport Cloud Platform is, by requirement, built using a cloud platform technology that offers a full set of services among which the MobiDataLab Project has selected mechanism to satisfy data consistency and persistence. Both persistence and consistence are implemented to be resilient and fault-tolerant: • At the infrastructural level: the platform is made of managed disks / storage known to be highly durable and available, simple and scalable for VM deployment and secured via Private Links. It also allows encryption of various categories, data disk storage up to32,767 gibibytes (GiB) and a maximum capacity of 4,095 GiB on the OS disk. • At the organisational level: a list of relevant data sources has been constructed (proposing relevant datasets) while harvesting. This list remains the last rescue solution if the platform has to be reconstructed and the harvesting to be performed ex-nihilo. Still back-ups, either handmade or managed by the cloud, are available as well as Images versus snapshots in the exploitation processes. Virtual Machine and storage cloning is another way to proceed to a quick re-creation of an identical ecosystem if needed. • At the application level: the open data world is constantly evolving and being enriched. Keeping that in mind is important. Therefore, it is necessary to frequently re-import and re-harvest the last updated set of information and sources to remain accurate and consistent with the offering given to the data consumers.

FR-GENERIC.02	Loader capability
Description	Data consumers and data providers may want or need to actively upload their datasets within the Transport Cloud -- indeed, in some scenarios they may not be able to provide their data through a service that they can expose.
Objectives	• The Transport Cloud must provide a **loader component** that allows data consumers and data providers to load the datasets they want or need to provide to the Transport Cloud.
Implementation	Organisation to business (O2B) requires the Transport Cloud Platform infrastructure to allow outsiders to import data via different mechanisms: • data imported from the partners are expected to be metadata as well as datasets intended for, potentially, customized implementations. The MobiDataLab project also offers journey planner functionalities that are built on General Transit Feed Specification data input associated with geographical map extract to be imported by the journey planner for visualization and more precise results. • services affected by partners import capability are the metadata services, which will have to allow identified outsiders to import and enrich the local metadata catalogue, as well as store data in relation to the description provided by the metadata. Mechanisms in the MobiDataLab project that allow to import data from external actors are the user interface and the metadata catalogue services. Both will also be exposed via APIs that handle the queries and produce consistent answers, regardless of the medium or interface used to make the queries.

FR-GENERIC.03	Metadata catalogue
Description	Actors interacting with the Transport Cloud may want to find, browse, and explore datasets. Such datasets can be available either within the Transport Cloud or from third party data providers.
Objectives	• The **metadata catalogue** available in the Transport Cloud must provide functionalities that allow to find, browse, and explore datasets that are of interest to the data consumer.
Implementation	The MobiDataLab project is built on a hybrid principle dictated by data ownership and licensing:

	• hosting metadata by the Transport Cloud and forwarding queries to browse datasets hosted by the owner whose address is described in the metadata. • Browsing metadata and querying datasets that are hosted within the MobiDataLab project infrastructure. Consequently, for each data use case the difference between local and remote dataset query or dataset import for local query will be a matter of legal ownership and/or data owner sovereignty. As such, the Transport Cloud offers an interface to browse the metadata catalogue and exposes detailed information about each metadata, in particular the location of the dataset(s) available at the address described in the associated metadata.

FR-GENERIC.04	**Service catalogue**
Description	Actors interacting with the Transport Cloud may want to find, browse, and explore the services provided by the platform or by third party service providers associated with the Transport Cloud.
Objectives	• The Transport Cloud must provide a **service catalogue** component.
Implementation	The Transport Cloud is made of not only processors and enrichment mechanisms, but it also offers services that must be presented to the data consumers/providers in the form of a catalogue of services so that they know which services they can use. For each service, the service catalogue provides: • the service name to help maintain consistency of the names to be used by all the data consumers/providers. • The service description, to get a human readable description of what to expect from the service. • The service API and parameters, to query the service from the data consumers'/providers' side. The full description of the service catalogue will come as a user-friendly section within the Transport Cloud documentation section followed by the API documentation intended for the third-party software programmers.

FR-GENERIC.05	**Basic Data Access API**
Description	Actors interacting with the Transport Cloud want to retrieve data from it, or from third party data providers that provide data to the Transport Cloud.

Objectives	• The **data API** must provide access to datasets that are of interest to the actors, either within the Transport Cloud or available from third party data sources, that **do not require any preliminary data processing** (e.g., anonymisation, enrichment, etc.) before being accessed.
Implementation	Among the set of functionalities offered by the Transport Cloud Platform there is a catalogue of services which includes basic data access service. The Transport Cloud Platform is intended to bring standardized data from a single-entry point that will gather both hosted and remotely stored data from third-party players. A dedicated family of API will give access to remote data from the Transport Cloud Platform, and to do so one must understand that remote data is out of the MobiDataLab jurisdiction and thus the actors accessing the data may face discrepancies such as: • data access capabilities due to network restrictions. • data access is limited to data owner registration policies. • data access restriction in terms of volume or number of queries. Furthermore, it is important to keep in mind that since data has not been processed nor converted into a Transport Cloud Platform exploitable format, it is not guaranteed that the resulting set of information will be of any use by the actors and that the data usability will remain under the actor's exploitation capabilities. The Transport Cloud Platform documentation will insist on such a use case to draw the developers' attention to the matter.

FR-GENERIC.06	**Advanced Data and Service Access API**
Description	Actors interacting with the Transport Cloud want to retrieve data, either from the Transport Cloud or from third party data providers that provide data to it, with the additional requirement that such data require appropriate processing (e.g., enrichment, anonymisation, etc.) before being accessed.
Objectives	• The **service API** must provide access to data sources containing datasets of interest to the data consumer and that **do require some preliminary data processing** (via a suitable processor) before being accessed.
Implementation	The MobiDataLab project understands the need for a set of APIs to access available data after the processing stage. Depending on the format of the dataset imported to the Transport Cloud Platform, a set of APIs are publicly exposed to target different type of hosted data: • data that are presented as raw datasets. • data are stored in a database service.

	• data are eventually indexed and accessible via the indexer query language. It is assumed that data which requires processing cannot be put to public use in order to comply with privacy and regulatory rules in place in the European Union.

FR-GENERIC.07	Converter API service
Description	Combining several third-party mobility data sources and services is among the main goals of the MobiDataLab platform. However, data and services provided by third parties are likely to follow different standards or conventions. The Transport Cloud must therefore provide a specific converter component aimed at standardising and reconciling different representation formats.
Objectives	• The data and service APIs must provide access to data represented with a homogeneous and standard data format. To this end the platform shall rely on a **converter component**, thus ensuring the data will be interoperable and standardised.
Implementation	The core existence of the MobiDataLab project is to offer mobility users a set of functionalities based upon public data coming from different sources using different data formats. The Transport Cloud therefore manages the presentation format as well as the data format during the importation and processing, in order to bring to the actors two different entry points with a standard that simplifies and harmonizes inter-connections: • Data acquisition is the first step in the global MobiDataLab workflow and consists in importing or harvesting data from providers who have open data or map/geographic data. • The data formats used for the presentation are mainly text files using well-established standards such as JSON, XML, and CSV, marked up with tags to be identified, parsed and formatted before being sent as a response to the users that query the Transport Cloud Platform. • The content of the data usable from storage is available with a wide range of information that the client/user is invited to navigate through via queries. The content of these databases managed by the MobiDataLab project is imported as standardised mobility/map data (e.g. GTFS, OSM, WFS, etc.) and made available by the Transport Cloud Platform when processed at the request of the client/user. When appropriate, conversion APIs can be used (e.g. the GTFS2NeTEx converter).

FR-GENERIC.08	Identity management
Description	Actors who want to interact with the Transport Cloud must have the appropriate credentials to access and interact with the platform.
Objectives	• The Transport Cloud must provide an **identity manager** component being able to authenticate the actors.
Implementation	Identity management is paramount to distinguish expected actors from assailants trying to access the Transport Cloud Platform. In order to secure the Transport Cloud Platform, it is necessary to divide identity management into different categories: • Application-oriented identity management hosted and managed by the application itself. The metadata service will allow to import metadata, and successively select metadata to be made public in order to support result accuracy for end data users. The access management is therefore from Organisation to Business, O2B, and will be managed by an administrator that will register users or sub-administrators entitled to register trusted users that will be granted partial or total prerogatives for managing the metadata service. • Infrastructure-oriented identity management that regulates the access as well as the rate control allowed per user or per type of queries. While metadata is subject to dual workflow from both outsiders and insiders, the infrastructure is totally exposed to the internet users, through the public gateway, to query the Transport Cloud Platform. Consequently, the Transport Cloud Platform encompasses a software application gateway that will be the entry point and will manage the query flow with: • An authentication mechanism authenticating users and granting access to the Transport Cloud Platform. • A rate control mechanism mitigating any risks of query flooding negatively impacting the Transport Cloud Platform performance.

5. Transport Cloud implementation

The MobiDataLab project aims to centralise access to data from various data sources, thus allowing users to browse and access data through a single entry-point. Data may be stored either in the Transport Cloud or within third-party infrastructures. Moreover, the Transport Cloud provides access to a plethora of different data formats that are relevant to the MobiDataLab project, for instance, structured data from relational databases (rows and columns), semi-structured data (CSV, logs, XML, JSON), unstructured data (documents, PDFs), and binary data (images, video).

The technologies exploited for the implementation of the MobiDataLab components are presented in the following subsections and summarised in Figure 3. A detailed description of each technological solution, and the reasons why they have been selected to implement the components of the Transport Cloud architecture, are detailed in the subsections.

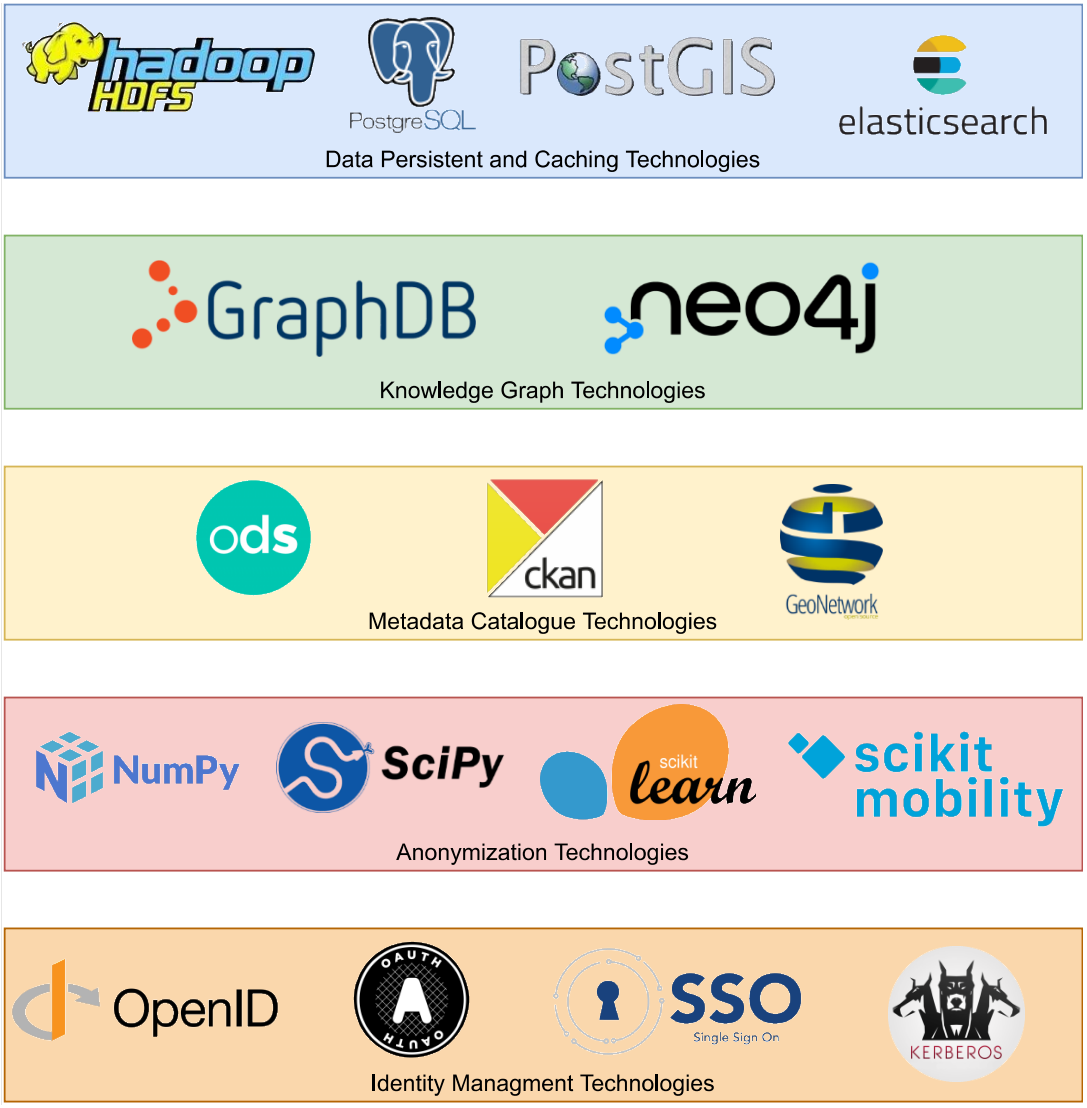

Figure 3: Candidate Technologies and Frameworks

5.1. API Gateways

Acting as an entry point to the information system, an API gateway is primarily a frontier entity between the client and the information backend. Requests come from the users in different formats in accordance with the user expectation: a user may expect timetables while another will be looking for baby cart compatibility, elevator for wheelchairs, etc. Each format is analysed by the API gateway before being forwarded to the processing unit hosted by the MobiDataLab platform. That processor response to the API gateway with the response to the initial query (electric car charger availability around the user GPS position, etc.). The API gateway replies to the client in a second phase.

The API gateway is at the frontmost point of the infrastructure and has the duty to welcome users, but also to face potential threats inherent to internet activity such as Denial of Service (DOS) by flooding the gateway with high volumes of traffic or software attacks by combining set of data in order to reveal potential vulnerability and take partially advantage of the platform.

A trade-off must be found between programming such interface that will satisfy the previously described requirements and, alternatively, the time spent in configuring a fully functional API gateway broadly used.

Many API Gateways solutions are available on the market. Table 1 shows a selection of the most popular ones:

Table 1: API Gateway Solutions

Framework	Pros	Cons
Kong	• Includes key authentication • Includes traffic rate limitation • Free use licence	• Poor Graphical User Interface (GUI) • Configured with REST commands
Tyk.io	• GUI interface • JavaScript Object Notation (JSON) like config language	• Fully licenced • Cloud Infrastructure as a Service (IAAS) and hosted by editor
Apigee	• Edge version from Google • Google account required	• Poorly supported • No software packaging • Poorly documented
Express Gateway	• Node/Express JavaScript (JS) friendly • Multiple modules needed → high cost • Red Hat Package Manager • (RPM) package ready • Microservices oriented • Key authorisation/rate limiter	• Dedicated shell (still easy configuration) • No GUI

	• Commercial support available	
Goku	• Design for large scale • Design for large volumes • Nice GUI • Identity management module • Load balancing customisation	• Curl configuration interface
Apache APISIX	• Design for large scale • Design for large volumes • Nice GUI • Identity management module • Load balancing customisation	• Curl configuration interface
Gloo	• Nice GUI • External authentication	• Kubernetes oriented • Dedicated Command Line Interface (CLI) • Partly under licence
Krakend	• Active development • Cloud and serverless oriented	• Commercial Licencing • IAAS hosted by the company → captive usage

The Transport Cloud project takes advantage of two different API gateway technologies in its design:

- The Azure **application gateway** that also plays a load balancer role. Included in all public cloud provider catalogues, this gateway offers a layer of protection between the internet and the **containerised application** as well as filtering policies to route queries. Finally, it provides load balancer functionalities to the request targets.
- The **Kong API Gateway** finds it legitimacy as an entry point in the instruction/request flow from the visitor to the **processors**. in addition to routing and filtering capabilities, the API gateway will offer rate control and authentication for incoming queries to protect the Transport Cloud plaftom infrastructure as well as the user experience.

5.2. Data Persistence Technologies

This section considers the technologies needed to enable the storage of data within the Transport Cloud.

The following paragraph therefore focus on solutions aimed at storing possibly large and heterogeneous data efficiently.

5.2.1. Database Technologies

PostgreSQL is a powerful, open-source object-relational database system with over 30 years of active development that has earned a strong reputation for reliability, feature robustness, and performance.

Elasticsearch is a distributed, RESTful search and analytics engine capable of solving a growing number of use cases. It is designed to provide excellent performance in different deployment settings, including an official Hadoop connector that will be integrated in the MobiDataLab Transport Cloud platform.

The database technology that makes the difference in the MobiDataLab project is by far PostgreSQL, not only for the worldwide reputation for decade, but also for its modularity when it comes to extensions among which, the PostGIS extension.

PostGIS is a spatial database extender for PostgreSQL object-relational database. It adds support for geographic objects allowing location queries to be run in SQL.

5.3. Knowledge Graph Technologies

In the context of the MobiDataLab project there is the need to store ontological and semantic information. To this end we will consider the industry-leading RDF triple store GraphDB from Ontotext. GraphDB is a semantic graph database fully compliant with the relevant W3C standards (i.e., RDF, OWL, SPARQL). GraphDB is designed to be highly efficient and robust at large-scale use cases. Moreover, it is one of the few triple store solutions that provide semantic inferencing, enabling its users to derive additional implied facts from the facts already extant in the store.

Another interesting technology we will consider is Neo4j[1], which is an open-source property graph store. This store supports atomicity, consistency, isolation, durability (ACID) transactions and has high-availability clustering for enterprise deployments. The store is accessible from most programming languages using its built-in REST web API interface, and a proprietary Bolt protocol with official drivers.

5.4. Metadata Catalogue Technologies

Metadata is a descriptive text file in which a data service presents the description of the available datasets content and format to the users; by many means, the metadata can be compared to a header or a presentation cover page that describes a whole set of serialised data. Basically, metadata can be seen as the data describing the dataset itself. Once the user/client

[1] https://neo4j.com/

is informed about the content of the metadata, that client is instructed about the location of the dataset, the type of data, as well as on the way to fetch this data from the entry point to query.

The metadata catalogue is a standard interface between a service and a user whose two steps approach is (1) to query the metadata service in order to get the list of available metadata and (2) to query the service and target a precise metadata file to get its content.

Spatial datasets require to be catalogued and described by up to date and publicly available metadata. Several metadata catalogues exist for the discovery of datasets and the sharing of the corresponding metadata. Among these, open-source solutions like OpenDataSoft[2], Comprehensive Knowledge Archive Network (CKAN[3]) and GeoNetwork[4]. Since these three solutions comply with metadata exchange standards, the MobiDataLab Transport Cloud can interoperate with each of them.

OpenDataSoft (https://www.opendatasoft.com/) is a cloud-based SaaS (Software as a Service) solution allowing users to publish, visualise and share data through tables and graphs. The data can be accessed via an API and a Data Hub. While this solution is private, its users can publish their open data publicly and limit the access to not public data within an organisation or a defined group of users, partners or employees with the help of its user-friendly management interface/ back office. Private companies such as Kisio Digital (Hove) and divers' cities (for instance Vancouver, the City of Paris and Newark use this solution as their main data catalogue which provides also a storage).

CKAN (https://ckan.org/) is an open-source tool used by data providers to publish their data and metadata through a web portal to make it more discoverable for users. CKAN is a widely used solution with an active community that make it quite flexible by developing useful external extensions, such as a CSW, RDF DCAT and INSPIRE harvesters. Unlike OpenDataSoft, hosting must be provided independently of the platform. CKAN itself is not specifically related to mobility data, but some transport authorities and many national access points use it as their open data portal.

GeoNetwork (https://geonetwork-opensource.org/) is a catalogue application for managing spatially referenced resources. It offers powerful metadata editing and searching features, an integrated interactive web map viewer and is based on open standards. It exposes metadata records through a Catalog Service for the Web (CSW) which makes it quite interoperable.

The two solutions retained to store and manage metadata for the Transport Cloud are CKAN and GeoNetwork. While OpenDataSoft requires less human maintenance in comparation to CKAN and GeoNetwork, CKAN is more interesting when frequent updates of many datasets are necessary and when data is not as standardized as in GeoNetwork (which is stricter in this matter). CKAN also can restrict, until a certain extent, the number of datasets of a particular thematic or type of organisation through its CKAN portal filters, which is not possible on GeoNetwork (other than in the search filter section). However, GeoNetwork deals better with the representation of geo-referenced data, and it has a large and well-established user specific GIS community, which might not be reached through more generic solutions like CKAN or OpenDataSoft. These two solutions were selected also as they could be federated into one as it was done by the OpenDataNetwork[5] Project which linked the

[2] https://www.opendatasoft.com/
[3] https://ckan.org/
[4] https://geonetwork-opensource.org/
[5] http://www.opendatanetwork.it/

geographical world with the alphanumeric open data. Therefore, it was more interesting to keep both solutions, which it further prevents to be lock-in into a single solution.

Harvesting is an important feature of metadata catalogues. Harvesting is the process of ingesting metadata from remote sources and storing it locally in the catalogue for fast searching. Harvesters provide a way for administrators to easily create and update an important number of datasets by importing them from an external source such as a Catalogue Service for the Web (CSW). The Data Catalog Vocabulary (DCAT) is an important standard supported by these catalogues. DCAT is "an RDF vocabulary designed to facilitate interoperability between data catalogues published on the Web"[6].

The MobiDataLab project aims to provide users with data from different stakeholders (e.g., public transport authorities, cities, regions, etc.) that may have different approaches to open data. As a result, some data sources will pass content through the MobiDataLab processing units while other data sources will be self-priding content to the data consumer. The MobiDataLab project will adapt to this paradigm by integrating metadata management within its data description strategy.

Each of these technologies are complementary and are adopted within the Transport Cloud Platform in the following way:

- OpenDataSoft as an open data provider
- CKAN as a metadata host and server leveraging multiple data format.
- GeoNetwork as a metadata host and server with GIS capabilities

5.5. Anonymisation Technologies

Privacy risk analysis builds upon statistical analyses on mobility data, including the similarity of positions and trajectories, their spatiotemporal distance, their unicity, and their autocorrelations through time.

On the other hand, the protection of positions and trajectories, whether following some privacy model such as k-anonymity[7] or differential privacy, entails their generalisation, their distortion by adding random noise, often from specific distributions, and/or their clustering based on their spatiotemporal distances.

Both statistical analyses and transformations can be supported by specialised numerical scientific analysis, as well as machine learning Python packages, such as Numpy[8], Scipy[9], and scikit-learn[10]. We further mention scikit-mobility[11], which includes utilities for synthetic mobility data generation,

[6] https://www.w3.org/TR/vocab-dcat-2/
[7] https://en.wikipedia.org/wiki/K-anonymity
[8] https://numpy.org
[9] https://scipy.org
[10] https://scikit-learn.org
[11] https://github.com/scikit-mobility/scikit-mobility/

privacy risk analysis for mobility data, and other general statistical analyses specifically targeting mobility data.

Regarding the generation of synthetic mobility data, we leverage sequence natural language processing (NLP) models. These models can be built using several frameworks for machine learning, such as Tensor Flow[12].

5.6. Identity Management Technologies

In this section we focus on the technologies needed to enable the Identity and Access Management (IAM) needs of the Transport Cloud.

The analysis is performed identifying main solutions for identity provisioning, authentication, authorisation and Single Sign-On.

Identity Provisioning. The most used standards for identity provisioning are the Lightweight Directory Access Protocol (LDAP), the System for Cross-domain Identity Management (SCIM). and the Service Provisioning Markup Language (SPML).

Authentication. The most widely used standards for Authentication are:

- **Kerberos**, a computer network authentication protocol which works on the basis of 'tickets' to allow nodes communicating over a non-secure network to prove their identity to one another in a secure manner;
- **OpenID connect**, a simple identity layer on top of the OAuth 2.0 protocol. It allows clients to verify the identity of the end-user based on the authentication performed by an Authorisation Server, as well as to obtain basic profile information about the end-user in an interoperable and REST-like manner;
- **Security Assertion Markup Language (SAML)**, an XML-based, open-standard data format for exchanging authentication and authorisation data between parties, in particular, between an identity provider and a service provider. SAML is a product of the OASIS Security Services Technical Committee.

Authorisation. The most widely used standards for Authorisation are:

- **XACML (eXtensible Access Control Markup Language)**," an OASIS standard that describes both a policy language and an access control decision request/response language (both written in XML). The policy language is used to describe general access control requirements, and has standard extension points for defining new functions, data types, combining logic, etc. The request/response language lets you form a query to ask whether or not a given action should be allowed and interpret the result. The response always includes an answer about whether the request should be allowed"[13];
- **OAuth**, an open standard for authorisation that provides client applications a 'secure delegated access' to server resources on behalf of a resource owner. It specifies a process for resource owners to authorise third-party access to their server resources without

[12] https://www.tensorflow.org
[13] https://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html

sharing their credentials. Designed specifically to work with Hypertext Transfer Protocol (HTTP), OAuth essentially allows access tokens to be issued to third-party clients by an authorisation server, with the approval of the resource owner. The client then uses the access token to access the protected resources hosted by the resource server. OAuth is commonly used as a way for Internet users to log into third party websites using their Microsoft, Google, Facebook or Twitter accounts without exposing their password. Current version is OAuth v2;

- **UMA (User-Managed Access)**, a profile of OAuth 2.0 defining how resource owners can control protected-resource access by clients operated by arbitrary requesting parties, where the resources reside on any number of resource servers, and where a centralised authorisation server governs access based on resource owner policies. Resource owners configure authorisation servers with access policies that serve as asynchronous authorisation grants.

SSO (Single Sign-On). SAML, OpenID Connect and Kerberos could be used to establish an SSO infrastructure as they define Identity Providers or Authentication Servers. Central Authentication Service (CAS) is another authentication protocol specifically targeted to SSO.

Although the choice of one or more of these solutions for the implementation of the Transport Cloud is not part of this version of the document, we can nevertheless make some assumptions. For instance, in an "API gateway scenario", OpenID Connect could be used to perform the initial authentication of the end user signing into the front-end login interface. This initial authentication includes an OAuth2 access token. When the request arrives at the API gateway, the access token must be extracted from the request, validated and "replaced with an access token whose scope matches the API provider's scope". (Broeckelmann, 2017)

Authentication wise, the Transport Cloud Platform needs to face the challenge of managing the authentication according to the following paradigms:

- The ideal topology being to have one directory listing all the users along with the authorisations and access for everyone listed in that directory. That unique and centralised directory will be the solution if the Transport Cloud component is offering common authentication options managed by a driver or standard functionalities widely spread in the industry. However, far from being the case, each component within the Transport Cloud has its own identity management system, they are incompatible each other and unable to share user validations. Thus, the alternative topology is the solution to adopt.
- The topology where each piece of component holds its own identification mechanism: each software includes within its core product, its identity management. This is the true nature of clusters made of autonomous technologies the MobiDataLab project had to dig into, to evaluate and to use individually for answering identity management requirements.

Consequently, it will be necessary to consider three layers of access having each a different set of access policies:

- APIs that are open to the public and eventually restricted by query rate controls that will be able to be more restricted with a prior registration in order not to put the platform under heavy load of requests.

- Partner services for which a generic user will be defined and shared with a group of users by the means of token or by network access filtering.
- The services managed by partners or data providers for which each user will be invited to self-register and will be granted access to its content.

5.7. Summary of Adopted Technologies

The extended set of technologies listed above were evaluated and a final selection of validated solutions is presented in the following table along with the use case or requirement for which the technology will match the expectation:

Technology	References	Selected	Functional requirement
API Gateway	3.2.2, 4.2.6(FR-F.01) 4.2.7(FR-G.01) 4.3(FR-GENERIC.05, FR-GENERIC.06, FR-GENERIC.07)	Kong	Dataset discovery Basic Data Access API Advanced Data and Service Access API Converter API service
Distributed File System Technologies	4.2.6(FR-F.03) 4.3(FR-GENERIC.0)	Azure storage services	Storage capability Data analysis
Database Technologies	4.2.7(FR-G.02)	PostgreSQL + PostGIS	Dataset combination and enrichment
Caching technologies	4.2.7(FR-F.02, FR-G.02)	Self-managed by processors	Dataset combination and enrichment
Knowledge Graph Technologies	4.2.8(FR-H.01, FR-H.02, FR-H.03)	GraphDB triplestore RDFLib (Python library) Ontologies that allow to represent semantically enriched mobility data	Dataset provision and discovery Dataset combination and enrichment Tourism analytics

Metadata Catalogue Technologies	3.2.1.1 4.2.7(FR-G.01) 4.3(FR-GENERIC.03)	CKAN + GeoNetwork	Metadata catalogue Dataset discovery
Anonymisation Technologies	3.2.3.2 4.1.1(NFR-GOV.02)	Developed by URV partner	Data anonymisation
Identity Management Technologies	4.3(FR-GENERIC.08)	Data catalogues application own functionality and infrastructure/API gateway management	Identity management

6. Other Promising Initiatives

As already highlighted in the Deliverable D2.6, the GAIA-X[14] project is the most relevant European initiative to the MobiDataLab project, as the former considers many aspects that are key to the realisation of the Transport Cloud. Compared to what we have reported in the first version of the Transport Cloud architecture dossier (Deliverable D4.1), it has to be noted that the GAIA-X project has progressed in several directions.

The GAIA-X Architecture document, which represents the most relevant document of the project, appears to have undergone several major updates. In the document's most recent version[15] , the authors provided further details on many high-level technical aspects that in the first version it was understood that were either treated preliminarily or left out. The aspects that have been added or further expanded mainly represent efforts to establish/adopt standards and mechanisms that allow to:

- Implement the notion of *policy*, i.e., statements of objectives, rules, practices, or regulations governing the participants within GAIA-X.
- Implement and use the notion of *self-descriptions*, which are data structures that follow the *W3C verifiable presentations* standard and that aim to describe verifiable claims concerning the entities participating in the GAIA-X conceptual model.
- Make GAIA-X ecosystems trustable and sovereign.
- Better define and implement the notion of "d*ata space*"[16] within the context of the GAIA-X project.

For what concerns the latter point, we report that the notion of data space within GAIA-X appears to represent a layer within one or multiple inter-connected GAIA-X ecosystems (which, we recall, will make up a multi-cloud federation) which goal is to provide a specific type of data (e.g., automotive, agricultural, tourism, medical, mobility) via data sharing and data cooperation among multiple participants. Such layer must implement the FAIR principles, as well as the principles of identity, trust, and sovereignty. This requires implementing the data space layer via several components. Among these, we report the *data space connector*[17], which is a key component that provides several fundamental capabilities to data space layers such as communication protocols, discovering, connecting, automated contract negotiation, policy enforcement, and auditing processes. Data space connectors must be able to communicate to each other.

[14] https://gaia-x.eu/
GAIA-X Architecture Document v22.04, released in April 2022, available at https://gaia-x.eu/wp-content/uploads/2022/06/Gaia-x-Architecture-Document-22.04-Release.pdf[15]
[16] https://gaia-x.eu/what-is-gaia-x/core-elements/data-spaces/
[17] https://github.com/eclipse-dataspaceconnector/DataSpaceConnector

Considering what has been said in the previous sections of this deliverable, we observe that while the notion of data space covers several aspects of interest to the MobiDataLab Transport Cloud architecture (plus additional aspects that reflect GAIA-X‘s broader scope), the Transport Cloud **is not just a data space** as it considers aspects that go beyond ”data”. Such aspects primarily concern the provision of services - indeed, we recall that the transport cloud aims to provide a wide range of services, and such efforts are reflected into the service API, service catalogue, and processor components. Also, data spaces require the data to be stored at the source (e.g., on the Association‘s members premises) rather than centrally. The Transport Cloud, on the other hand, does not impose this strong requirement, which in turn enables to store data within the architecture whenever needed. Finally, we conclude by saying that one day it may well be in the interest of the MobiDataLab project to connect the Transport Cloud to GAIA-X data spaces dealing with mobility data. This, in turn, would require making the Transport Cloud compliant with the GAIA-X’s architectural requirements. We observe, however, that GAIA-X is still at an early stage of development, hence we deem that at the present moment the most reasonable thing to do is to continue monitoring the project’s evolution

The GAIA-X project seems also to have made substantial progresses for what concerns the prototypes of the components that one day will be part of the GAIA-X architecture. To this end, we report that the GAIA-X project so far held four hackathons that focused on (1) increasing the technical competence and knowledge related to technology relevant to the GAIA-X ecosystem, (2) the integration and alignment of different ideas, concepts, pilots and prototypes to consistent approaches, and (3) the implementation of a few components of the architecture.

Finally, we report that we have found 13 GitHub public repositories[18] related to the GAIA-X project. Most of these repositories appear to be testbeds, demos, and examples concerning specific aspects of the architecture. A few other repositories, however, contain component prototypes that may be relevant to MobiDataLab. Such repositories are:

- The GAIA-X SCS Identity and Access Management (IAM) component (https://github.com/SovereignCloudStack/testbed-gx-iam)
- A minimal viable Gaia-X Catalogue (https://github.com/deltaDAO/mvg-catalogue)
- The Eclipse Dataspace Connector (https://github.com/eclipse-dataspaceconnector/DataSpaceConnector)

Overall, we deem that these prototypes are still at a too early stage of development to be integrated within the MobiDataLab Transport Cloud. We will, however, continue to monitor the evolution of the GAIA-X project.

[18] https://github.com/topics/gaia-x

Funded by the European Union

7. Conclusions

This document describes the architecture of the Transport Cloud platform supporting the data federation to enable the European-wide data sharing and trans-national access to the information provided by the federated repositories.

Starting from the requirements defined on a per use-case level (see Deliverable D2.9), we proceeded to define the actors expected to interact with the platform and how the information flows through it, the functionalities that the Transport Cloud needs to implement, and the requirements on a per-platform level. This then allowed to provide the specifications of the Transport Cloud architecture and to focus on the technologies needed to implement a prototype of the platform.

This document should be seen as one of the two pillars of the WP4, since it specifies which components are present in the transport cloud, their roles within the platform, the functionalities they must provide, and how they relate to each other. The second pillar is represented by the set of 4 demonstrators, which represent the software counterparts of the FAIR principles and concretely implement functionalities in relation to data catalogue (Task 4.2), data access services and data channels (Task 4.3), data processors (Task 4.4), and anonymisation and data privacy (Task 4.5).

Once put together, the two aforementioned pillars will allow to build a prototype of the Transport Cloud, which will be then used and validated by the participants in the Living and Virtual Labs, and continuously improved in this context.

Funded by the European Union

8. Bibliography and References

Broeckelmann, R. (2017). *Identity Propagation in an API Gateway Architecture*. Retrieved from https://cloud.google.com/blog/products/api-management/identity-propagation-in-an-api-gateway-architecture

MobiDataLab consortium

The consortium of MobiDataLab consists of 10 partners with multidisciplinary and complementary competencies. This includes leading universities, networks and industry sector specialists.

@MobiDataLab
#MobiDataLab

https://www.linkedin.com/company/mobidatalab

For further information please visit **www.mobidatalab.eu**

MobiDataLab is co-funded by the EU under the H2020 Research and Innovation Programme (grant agreement No 101006879).