Data-driven and Dynamic Space and Assets for Physical Internet-led Urban Logistics and Planning

Deliverable 3.1

DISCO Data Space open software repository Version 1

Panos Protopapas, INLE

17/07/2024

Funded by the European Union

This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101103954. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or CINEA. Neither the European Union nor the granting authority can be held responsible for them.

Table of Contents

Abstract

The Urban Freight Data Space (UFDS) aims to revolutionize urban logistics landscape by addressing the current fragmented state of data sharing in the industry. This deliverable outlines the foundational architecture and development plan for the UFDS, aiming to facilitate data integration and collaboration.

With urban freight logistics having grown into a complex and competitive ecosystem, the lack of comprehensive data sharing has led to operational inefficiencies and non-optimal use of cities' space that UFDS seeks to overcome by enabling secure, standardized data exchange while supporting and ensuring data sovereignty for all contributors.

More specifically, our project targets the creation of a robust, future-proof platform that will integrate city-owned datasets and progressively incorporate private data as technology advances. The deliverable details the strategic approach, including the technical development plan which will facilitate access to critical city-related data and support the creation of meaningful collaborative use cases. Our ultimate goal is to establish a data space that optimises logistics operations, reduces environmental impact, and enhances urban sustainability.

Summary sheet

Deliverable No.	
Project Acronym	DISCO
Full Title	DATA-DRIVEN, INTEGRATED, SYNCHROMODAL, COLLABORATIVE AND OPTIMISED URBAN FREIGHT META MODEL FOR A NEW GENERATION OF URBAN LOGISTICS AND PLANNING WITH DATA SHARING AT EUROPEAN LIVING LABS
Grant Agreement No.	101103954
Responsible Author(s)	Panos Protopapas (INLE), Thomas Kusmirczak (IMEC)
Peer Review	IMEC, IDSA
Quality Assurance Committee Review	FIT
Date	31/07/2024
Status	Final
Dissemination level	Public
Version	1
Work Package No.	3
Work Package Title	The Urban Freight Data Space for NetZero cities
Programme	Horizon Europe Innovation Actions
Coordinator	FIT CONSULTING SRL
Website	https://discoprojecteu.com/
Starting date	01/05/2023
Number of months	42

Project partners

Organisation	Country	Abbreviation
FIT CONSULTING SRL IT Coordinator	IT	FIT
RUPPRECHT CONSULT-FORSCHUNG & BERATUNG GMBH	DE	RC
INLECOM INNOVATION ASTIKI MI KERDOSKOPIKI ETAIREIA	EL	INLE
PNO INNOVATION SL	ES	PNO
INTERNATIONAL DATA SPACES EV	DE	IDSA
FM LOGISTIC IBERICA SL	ES	FM
AKKA INDUSTRY CONSULTING GMBH	DE	AKKA
FONDAZIONE ISTITUTO SUI TRASPORTI E LA LOGISTICA	IT	ITL
ETHNIKO KENTRO EREVNAS KAI TECHNOLOGIKIS ANAPTY XIS	EL	CERTH
LINDHOLMEN SCIENCE PARK AB	SE	LSP
KUHNE LOGISTICS UNIVERSITY GGMBH	DE	KLU
INSTITUT DE RECHERCHE TECHNOLOGIQUE SYSTEM X	FR	IRTX
STICHTING BREDA UNIVERSITY OF APPLIED SCIENCES	NL	BUAS
POLIS - PROMOTION OF OPERATIONAL LINKS WITH INTEGRATED SERVICES, ASSOCIATION INTERNATIONALE	BE	POLIS
EUROPEAN PARKING ASSOCIATION EPA EV	DE	EPA
ALLIANCE FOR LOGISTICS INNOVATION THROUGH COLLABORATION IN EUROPE	BE	ALICE
ERASMUS CENTRE FOR URBAN,PORT AND TRANSPORT ECONOMICS BV	NL	ERASMUS

INTERUNIVERSITAIR MICRO-ELECTRONICA CENTRUM	BE	IMEC
VLAAMS INSTITUUT VOOR DE LOGISTIEK VZW	BE	VIL
FUNDACION ZARAGOZA LOGISTICS CENTER	ES	ZLC
BE-MOBILE	BE	BE-MOBILE
STAD GENT	BE	GENT
OPLEIDINGSCENTRUM VOOR HOUT EN BOUW VZW	BE	OHB
CITYLOGIN IBERICA SL	ES	CITYLOGIN
UNIVERSITAT POLITECNICA DE CATALUNYA	ES	UPC
AJUNTAMENT DE BARCELONA	ES	BCN
VENICE INTERNATIONAL UNIVERSITY	IT	VIU
FUNDACION DE LA COMUNIDAD VALENCIANA PARA LA INVESTIGACION, PROMOCION Y ESTUDIOS COMERCIALES DE VALENCIAPORT	ES	VPF
FUNDACION DE LA COMUNITAT VALENCIANA PARA LA PROMOCION ESTRATEGICA EL DESARROLLO Y LA INNOVACION URBANA	ES	LAS NAVES
T-BOX DELIVERY & SOLUTIONS SL	ES	T-BOX
AYUNTAMIENTO DE ZARAGOZA	ES	ZARAGOZA
FUNDACION ZARAGOZA CIUDAD DE CONOCIMIENTO	ES	FZCC
FORUM VIRIUM HELSINKI OY	FI	FVH
KOBENHAVNS KOMMUNE	DK	COPENHAGEN
REGION HOVEDSTADEN DK Partner	DK	REGIONH
COMUNE DI PIACENZA	IT	PIACENZA
MESTSKA CAST PRAHA 6 / District Prague	CZ	PRAHA
REGIONAL MANAGEMENT NORDHESSEN GMBH	DE	RMNH

AARHUS KOMMUNE	DK	AAKS
DIMOS THESSALONIKIS	EL	THESSALONIKI
DIETHNIS EKTHESI THESSALONIKI AE	EL	TIF HELEXPO
ACS TACHIDROMIKES IPIRESIES MONOPROSOPI ANONYM	EL	ACS
ROLAN OY	FI	ROLAN
ASOCIACIÓN LOGÍSTICA INNOVADORA DE ARAGÓN	ES	ALIA
A to B Finland Oy	FI	A2B
GETPLUS srl IT Partner	IT	NEXT
COMUNE DI PADOVA IT	IT	ComPADUA

Document history

Version	Date	Organisation	Main area of changes	Comments
0.1	02/05/2024	INLE	Table of Contents	
0.2	15/05/2024	INLE	Sections 1.2, 1.3, 4.1	
0.3	16/05/2024	IMEC	Sections 1.2, 2.1 2.2, 2.3	Sections not complete, diagrams need to be added on the working of every component
0.4	23/05/2024	INLE, IMEC	Section 3	WIP, added detailed information on the components, added architecture diagram for V1
0.5	28/05/2024	IMEC	Section 2	Architecture and related information

0.6	30/05/2024	INLE	Sections 3.2, 4	
0.7	03/06/2024	INLE	Section 2	Identity Management, App and Connector Store
0.8	06/06/2024	IMEC	Sections 2, 3	Federated catalog & Broker, Gent LL info
0.9	18/06/2024	IMEC	2.2.2, 2.4, 3.1.1	DISCOLLECTION updates across the different sections. Updated description of the Gent datasets
0.10	18/06/2024	INLE	2.1, 2.2, 2.3.1, 2.3.2, 2.3.3	Architecture
0.11	20/06/2024	INLE	2.3.4, 2.3.4	Architecture
0.12	25/06/2024	INLE	3.1, 3.2	Implementation
0.13	27/06/2024	INLE, EPA	Appendix	Appendix
0.14	15/07/2024	INLE	Abstract, Section 1.1. and 5	
1.0	16/07/2024	INLE	Final Edits	Panos Protopapas Antonis Mygiakis Ioanna Fergadiotou
1.1	17/07/2024	INLE	Addressing IDSA comments	
1.2	18/07/2024	INLE	Inclusion of final comments from IDSA, IMEC, FIT	**The final version is ready to be submitted**

List of acronyms

UFDS	Urban Freight Data Space
FOSS	Free and Open Source Software
UI	User Interface
LSP	Logistics Service Partners
IAM	Identity and Access Management
OIDC	Open ID Connect
CAs	Certificate Authorities
DAPS	Dynamic Attribute Provisioning Service
ParIS	Participant Information Service
RAM	Reference Architecture Model
EDC	Eclipse Dataspace Connector
EDR	Endpoint Data Reference
SDP	Smart Data Platform
IdP	Identity Provider

1. Introduction

In the bustling roads of our cities, a revolution is underway within the urban freight industry. Once consisting solely of delivery trucks shuffling packages, this sector has experienced significant growth in recent years. Driven by the surge in last-mile logistics and the proliferation of delivery services, it has transformed into a dynamic ecosystem including last-mile logistics, cross-town deliveries, and out-of-town hubs.

However, the current state of data sharing in the industry is fragmented. Urban logistics operations are naturally distributed across a multitude of areas, actors, and locations, all requiring up-to-date data to synchronize planning and monitor execution. Today, companies primarily rely on their own datasets, limiting their ability to optimize logistics operations comprehensively.

This is where the concept of data spaces comes into play. The Urban Freight Data Space (UFDS) aims to support data sovereignty for all its contributors, ensuring they maintain control over their data while benefiting from shared insights. By facilitating a secure and standardized environment for data exchange, UFDS can help all stakeholders access critical city-related data, collaborate more effectively, and optimize their operations, ultimately leading to more efficient, sustainable, and greener urban logistics solutions.

Our project goals are to establish a solid foundation for this innovative data-sharing ecosystem, highlight the advanced possibilities it offers, and document further opportunities for the future. This deliverable focuses on the architecture of the UFDS, outlining our objectives and development plan for the project.

1.1. How the Urban Freight Data Space revolutionises urban logistics

The urban freight industry has witnessed a significant transformation, driven by the surge in last-mile logistics and the proliferation of delivery services. This growth has transformed it into a dynamic ecosystem including last-mile logistics, cross-town deliveries, and out-of-town hubs. This expansion has led to a dynamic and competitive market, producing a complex pattern of movement on a daily basis by countless competing companies.

This expansion has led to a dynamic and competitive market producing a complex pattern of movement on a daily basis undertaken by countless competing companies. On the one hand, this is a testament of a healthy industry with competition breading innovation, but on the other hand it feeds a chaotic situation with all actors operating largely within their own data silos, in perfect knowledge of their side of the data (trucks' locations, rhythm of their warehouses, evolution of their delivery routes, etc.) but at the same time with decreased access to other important information they are highly dependent on, like the flow of traffic, parking availability, and other environmental factors, ultimately introducing challenging to the city-space as a whole in the form of urban congestion and related inefficiencies.

Unfortunately, the current state of data sharing in the industry is fragmented although urban logistics operations are by nature distributed between a multitude of areas of activity, actors, and locations, all requiring up-to-date data by all parties, to synchronize planning and monitor the execution, including the needs for replanning. That is, companies primarily rely on their own datasets, limiting their ability to optimise logistics operations comprehensively although scenarios in which data would be allowed to flow and transcend corporate silos would allow for the full potential of urban freight logistics to be reaped, optimising not just the efficiency of different private actors but -more importantly- the city's well-being.

The concept of data space, particularly the Urban Freight Data Space (UFDS), seeks to address these challenges by promoting data sovereignty for all its contributors This ensures that participants retain control over their data while gaining shared insights within a secure and standardized data exchange environment that provides them with access to critical city-related data, enabling them to collaborate more effectively and optimise their operations, leading to more efficient, sustainable, and greener urban logistics solutions. Specifically, the UFDS developed and demonstrated in DISCO will drive a paradigm shift in operations by:

a) Enhancing data sharing and collaboration among various stakeholders in urban logistics, including transport operators, city planners, retailers, and delivery services. Data sovereignty mechanisms will ensure that operations are governed by trust, leading to better coordination and collaboration, reducing inefficiencies, optimising routes and space management, and improving the overall flow of goods within urban areas. The DISCO UFDS aspires to become a reference digital enabler for the deployment of data-driven collaborative planning services. This will be achieved by providing a trustworthy framework for sharing data related to transportation, delivery, space use, capacity and supply chain operations in urban areas across decentralised networks, whilst introducing semantic interoperability. The latter is particularly critical for establishing a common understanding of data meaning and context across diverse sources and applications. Through the DISCO UFDS data sovereignty, transparency and fairness are ensured, creating the ground for stakeholders' onboarding. Data connectors can facilitate seamless data exchange services, while incorporating robust data security enforcement mechanisms.
b) Making feasible the development of decision-making applications that utilise integrated data. By integrating diverse data sources into the UFDS, urban logistics can benefit from comprehensive insights, analytics and even AI applications to predict demand, optimise resource allocation, and manage city logistics networks more effectively. Further, this capability can support the development of innovative collaborative business models.
c) Contribute positively to the EU sustainability agenda and support the development of sustainable urban logistics strategies. The UFDS can play a crucial role in promoting sustainable urban logistics by enabling better tracking and management of environmental

impact. Through the aggregation and analysis of data on vehicle emissions, energy consumption, and delivery patterns, stakeholders can identify opportunities to reduce carbon footprints, implement greener practices, redesign the use of space for more liveable cities and comply with environmental regulations.

In summary, in DISCO we aim to establish a solid foundation for this innovative data-sharing ecosystem, highlight the advanced possibilities it offers, and document the further possibilities that can be offered in the future. By doing so, we aim to showcase how UFDS can revolutionize urban freight logistics, driving efficiency and sustainability through enhanced data integration and cooperation. In this deliverable, we focus on the architecture of the UFDS to achieve these goals and outline our objectives and development plan for the project.

1.2. Development plan

Based on the vision set above, we have decided on a strategic approach capable of addresses the needs of the project and succeed in building a robust and future-proof platform. The approach takes into account the project's capabilities, tackles the main challenge of the project described below, and aims to deliver a platform fit for the anticipated requirements of an urban freight data space.

This challenge is the reluctance of the private data owners to share their data via the UFDS due to the not yet fully developed state of the data space technology, which does not reliably support the level of data sovereignty required by these stakeholders. On the bright side, a substantial amount of city-owned data will be incorporated into the data space. The datasets that will be incorporated into the data space are discussed throughout Section 3.1.1.

Our approach however does not merely plan to develop a data space using currently available technical components which will incorporate only the datasets available to us. It is also designed with future technical advancements in mind, ensuring that the platform will be capable of meeting the evolving needs of private data owners as the data space technology matures. Specifically, we plan to address these issues via three Targets discussed below and by making use of relevant ontologies when describing different datasets (see Section 2.3.5 on the Vocabulary Hub component and Appendix: APDS Ontology).

Target 1: We want to gather some of the city datasets into a cohesive first version of the UFDS which will provide users with a functional platform to view and access this data; in the rest of the deliverable, we refer to this milestone instance of the data space as UFDS Version 1, it will be deployed in October 2024. A report of this version will be provided in Deliverable D3.2 – UF Data Space Connector Store.

Target 2: Following this initial deployment our second target is to collaborate with our partners in WP2 and WP4 to finalize a few meaningful use cases for the UFDS, tailored to leverage the integrated data effectively; moreover, we plan to explore the potential data synthesis methods to also support scenarios including dataset types that won't be otherwise available (e.g., privately owned datasets). These use cases and datasets will be complemented by further technical development of the components of UFDS Version 1 in order to incorporate all further functionality required to support them; in the rest of the deliverable, we refer to this milestone instance of the data space as UFDS Version 2, it will be deployed under a rolling update model throughout 2025 via software updates that will become available on the Open Software Repository (see Section 4). UFDS Version 2 will serve as a foundation for a future ecosystem where all urban logistics datasets (i.e., both public and private) can be accessible. An update of this version will be provided in Deliverable D3.2 – UF Data Space Connector Store while a report in Deliverable D3.4 – DISCO Data Space open software repository Version 2 and Implementation Guide.

Target 3: Our final target is to document the needed advancements, further than what will be technically achieved by UFDS Version 2, in order to establish an urban freight data space that will become integral to the future landscape of urban logistics data exchange, meet the needs of private data owners, and offer its users the envisioned usability that the DISCO-X meta models have set out for in full. This documentation will be provided in Deliverable D3.4 – DISCO Data Space open software repository Version 2 and Implementation Guide and will offer technical insights and guidelines for post-DISCO advancements using UFDS version 2 as the groundwork.

1.3. Relation with the rest of DISCO

This section outlines the position of Work Package 3 within the broader context of the DISCO project and specifically its interrelations with work packages 2 and 4, part of an iterative process underlying the development of the UFDS.

On one hand, WP2 has been instrumental in identifying and describing beneficial uses of the data space by the prospective users of the UFDS, presented via the DISCO-X meta models, detailed in Deliverable D2.1 – Urban Logistics Transition Requirements and to be further elaborated in subsequent deliverables. Essentially, these meta models provide high-level, generalized blueprints of data space applications, guiding the development of use cases that the UFDS aims to support.

On the other hand, WP4 and the Starring Living Labs contribution, in terms of technical input, is the preparation and sharing of datasets either openly accessible or on the path to becoming so; datasets that can be integrated into the UFDS despite the current limitations in data sovereignty capabilities of the data space technology mentioned above (Section 0). Moreover, another crucial aspect is the drafting of meaningful use cases for the data space, a process that is heading towards completion in the coming months following the process discussed in Section 3.2.

WP3's role lies between WP2 and WP4, with its tasks being to develop a data space that incorporates the necessary datasets and functionality to enable the development of applications aligned with the high-level blueprints of the DISCO-X meta models. This requires a dual focus: addressing the current technical limitations of data space software offerings and designing a system that can evolve to meet future needs.

This will be achieved through an iterative process involving on one side the DISCO-X meta models from WP2 and the design of technical blueprints for applications based on their "spirit", that could utilise available datasets and lead to practical and valuable use cases for the Living Labs of WP4, and on the other side seeking feedback and collaboration from WP4 of the practical applicability of these conceptual designs, as well as on the availability of fitting datasets for said designs. This dynamic interaction aims to bridge the gap between theoretical models and practical implementation ensuring that the UFDS will not only meet current needs but will also be adaptable for future enhancements.

Specifically, and as mentioned in Section 3.2, this process will likely necessitate the creation or synthesis of additional datasets to cover the needs of prospective applications. Moreover, WP3 will deliver a comprehensive report on the technical advancements required to develop a data space infrastructure that meets the needs of all its foreseeable users. This report will be included in Deliverable D3.4 – DISCO Data Space open software repository Version 2 and Implementation Guide.

1.4. Document structure

The structure of the rest of the deliverable is as follows.

In Section 2 we begin by introducing the reader to the concept of data spaces through an overview of the International Data Space Association's (IDSA's) reference architecture (Section 2.1); then, we present at a high-level the UFDS's architecture explaining any departures from the reference architecture (Section 2.2) differentiating between the UFDS Version 1 and UFDS Version 2; next, the components of the UFDS are examined in Section 2.3; finally, we present and discuss DISCOLLECTION (Section **Error! Reference source not found.**), which although developed as part of WP2 (T2.3.5 – DISCO-X Meta Model Suite innovations) and planned to be further elaborated in Deliverables D2.3 (DISCO Meta Model Suite architecture) and D2.5 (DISCO-X & Meta Model Suite Interoperability), it is technically integrated into the UFDS's "extended" architecture.

Section **Error! Reference source not found.** explores the implementation of the UFDS, detailing for UFDS Version 1 the relevant use cases, elaborating on the datasets to be incorporated and its expected users across the different Living Labs (Section 3.1); this is in line with what has been defined as our first target (Target 1) in Section 0. Then, we detail the next steps to be taken in accordance to our next targets (Targets 2 and 3, Section 0) in order for these to be achieved (Section 3.2) and to submit the necessary documentation of the advancements required to establish an urban freight data space that will become integral to the future landscape of urban logistics data exchange in Deliverable D3.4 (see Section 3.2).

Section 4 describes the open software repository which will offer users easy access to the technical components of the UFDS, including applications developed for it. Finally, Section 0 summarizes the deliverable and the next steps to take towards achieving our goal.

2. Architecture

In this section we start by introducing the reader to the concept of data space, *a secure, sovereign system of data sharing in which all participants can realize the full value of their data*[1] (Section 2.1). To do so, we present an overview of IDSA's reference architecture, its merits and capabilities, and briefly explain the role of the different components included. Next, we present at a high level the adaptation of this architecture for the needs of the DISCO project (Section 2.2), followed by a detailed technical view for each component (Section 2.3). Finally, DISCOLLECTION Meta Model is discussed in Section **Error! Reference source not found.**; although not technically part of the UFDS, DISCOLLECTION is planned to act as a multi-purpose middleware between UFDS and some of the available datasets and as such, its role is presented here.

2.1. High-level overview of IDSA framework

Data spaces, as defined by the International Data Spaces Association (IDSA), represent a revolutionary approach to data sharing and monetization, ensuring that companies can securely exchange data under a framework of trust and standardized rules. Based on the concept of a decentralised data exchange market, the IDSA's architecture promotes data sovereignty, allowing entities to maintain control over their data while enabling innovative collaborations and lowering entry barriers in the data economy. Certified data providers and recipients operate within these data spaces, fostering a trusted ecosystem where participants can seamlessly share valuable data. By leveraging uniform standards and governance, data spaces pave the way for limitless innovation and cooperative ventures, transforming the future of data exchange.

A central component of the IDSA architecture is its emphasis on trust, security, and decentralized data storage. Each participant undergoes rigorous evaluation and certification, ensuring the integrity and security of the data exchanged. Data sovereignty is upheld through strict usage policies that data consumers must adhere to, maintaining the data owner's control. The architecture supports real-time data search via Metadata Brokers and promotes standardized interoperability

[1] Source: https://internationaldataspaces.org/

through the Data Space Connector. Additionally, the system allows for the integration of value-adding applications, enabling sophisticated data processing and analytics services. This comprehensive framework not only facilitates the creation of new, data-driven business models but also supports a dynamic ecosystem where data is a key strategic resource, driving joint innovation and value creation among its members.

An overview of the high-level reference architecture is shown in Figure 1 below. It depicts an exchange of data between two participants on the data space, the Data Provider and the Data Consumer. Loosely speaking, the two participants identify themselves on the data space via the Identity Provider, then the Consumer can find the data offering of the Provider via the Broker, and connect Connector-to-Connector with them in order to retrieve the data needed but only after accepting the Provider's Usage Policies on the requested datasets; note that these Usage Policies could altogether bar the Consumer from ever receiving this data. Moreover, the "language" of the data (variable names, value limits, etc.) can be standardized via the Vocabulary component, while any pre-processing (Provider side) or post-processing (Consumer side) of the data can be achieved via applications provided by the App Store. Finally, all described actions on the data space can be recorded by the Clearing House which can resolve various disputes, facilitate payments, etc.

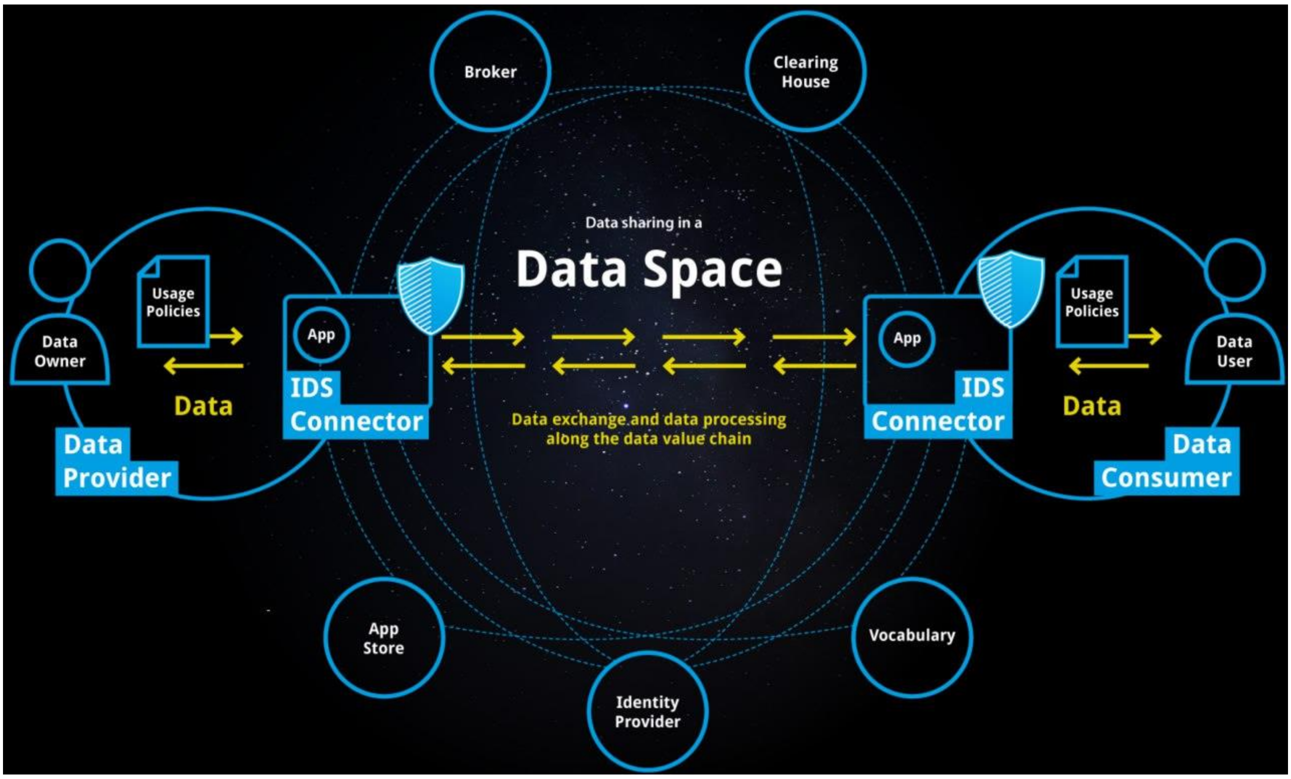

Figure 1: Data spaces high level reference architecture (Source: https://internationaldataspaces.org/why/data-spaces/)

2.2. Adaptation of IDSA's framework to DISCO's content (UFDS)

When setting to adapt the IDSA's framework presented above (Section 2.1) for the purposes of the DISCO project and the needs of urban logistics, in order to achieve our technical development goals, from a technical standpoint, we have set three base design principles:

1. **Prioritise existing -open source- technology** as we need to exploit existing data space technology components and use them as building blocks to abstract workflows implementing data discovery and data transfer, and only apply the necessary development refinements where needed. Furthermore, the open source nature of these building blocks will allow us to open source the building blocks that will compose the UFDS.
2. **Enable user interaction through a UI** in order to develop a user-friendly portal, where UFDS participants become part of a community of data producers, intermediaries, and data consumers. This portal(s) masks the complexity of the data discovery and research workflows, helping participants in establishing and managing data sharing agreements and their data usage policies, as well as find and use relevant applications aiding them in performing their intended data usage.
3. **Use "general release" level mature technologies** so that we minimize regression risks associated with rapidly evolving data space standards, by relying on technologies and protocols having reached a "general release" level of maturity. A consequence of this is that not all desired features might be available initially. On the other hand, more "mature" software releases are expected to be relatively "bug-free", somethding important for the development cycle of the UFDS, since data spaces are a novel technology and related software is not expected to be fully tested for stability and reliability.

Based on the above principles, we have investigated the available data space software packages and identified two viable solutions:

- The Eclipse Dataspace Connector (EDC), an open-source project developed under the Eclipse Foundation. It is a comprehensive framework including a comprehensive set of concepts, architectural elements, code, and examples, offering functional and non-functional features that can be reused and customized by developers and ensure interoperability by design. It is a framework following the specifications of the IDSA Data Space protocol mentioned above (Section 2.1) as well as that of Gaia-X.
- The Sovity Connector built upon the Eclipse Dataspace Connector and leveraging its robust framework. It offers a secure, scalable, and flexible solution for data sharing and integration, while maintaining data sovereignty and security. Compared to EDC, it offers a relatively easier and more user-friendly deployment process albeit more opinionated. This is attributed to Sovity's dual-tier approach, including both a FOSS tier (which we have identified) and a non-FOSS tier aimed at fostering development and attracting a broader user base.

Not only are both solutions open-source, mature, and offer graphical interfaces, they both share the same underlying codebase (i.e., EDC's codebase); this implies that if the need to switch from one connector to the other during the project arises, the migration process should be straightforward thanks to the common codebase and the direct transferability of the experience the technical team will have gained during development.

2.2.1. UFDS High-level architecture

As mentioned above in Section 0, our first target in our development plan is to deliver UFDS Version 1 by October 2024. This initial release will provide users with access to three datasets from the Living Labs of Ghent and Thessaloniki. These datasets, described in Section 3.1.1, align with three DISCO-X measures: DISCOCURB, DISCOPROXI, and DISCOESTATE. Two of these datasets will be transformed into the EPA/APDS standard via DISCOLLECTION, with further details on DISCOLLECTION 's functionality provided in Section **Error! Reference source not found.**. The structure and purpose of the EPA/APDS ontology are elaborated in Appendix: APDS Ontology.

The datasets focus on parametrizing and visualizing geospatial data. To demonstrate these use cases, a proof-of-concept visualization app, DISCOEYE, will be developed. This app will be integrated with UFDS, enabling users to dynamically visualize the datasets on a map. The adherence to EPA/APDS and Datex II standards will facilitate future customizations of DISCOEYE to support additional datasets compliant with these standards. Moreover, given the geospatial nature of many datasets and the DISCO-X measures' emphasis on geospatial use cases, DISCOEYE has the potential to serve as a foundational software for future UFDS applications.

On their side, the Living Labs will utilize a Smart Data Platform (SDP) hosted by the city of Ghent, through which the city will upload their datasets. This SDP will connect to the data space via a UFDS Connector. DISCOEYE will retrieve and filter data from the data space through the SDP for Ghent, and directly for Thessaloniki, to visualize it on a suitable map projection.

To support these functionalities, UFDS Version 1 will require several key components: the Connector, Metadata Broker, and Identity Management components. Detailed information on the SDP, DISCOEYE, and the essential data space components can be found in Sections **Error! Reference source not found.**, 2.3.4, and 2.3, respectively.

We illustrate the architecture of UFDS Version 1 in Figure 2.

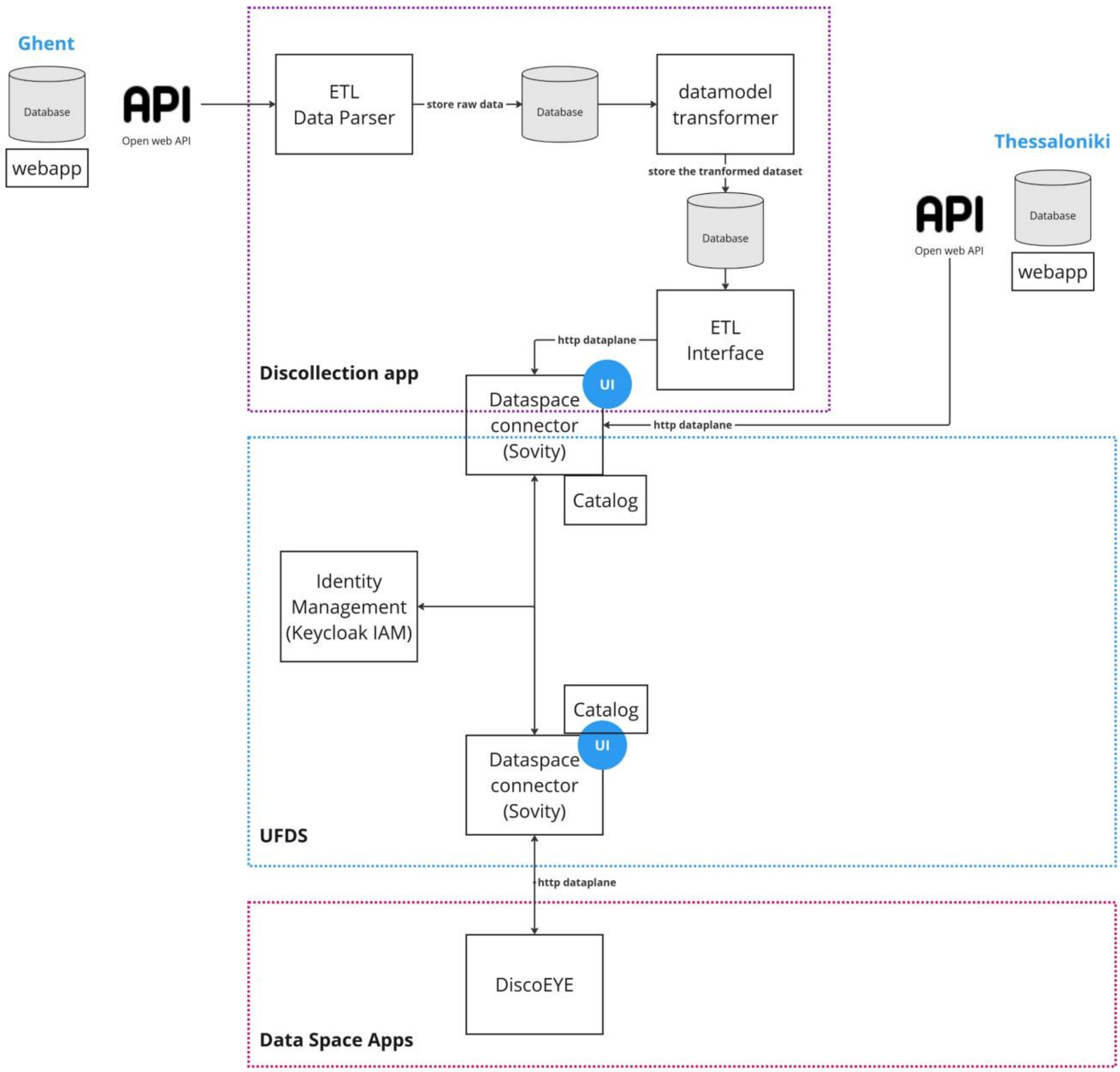

Figure 2: UFDS Version 1 high-level architecture

Following the release of UFDS Version 1, WP3 will not only update the initial components as needed, but also extend UFDS's functionalities with the required features that will support the developed use cases and supporting data space applications. Currently, we envision these needs to require a Vocabulary Hub, allowing users to explore the main vocabularies of the data space (EPA-APDS, Datex II) and their links with the available datasets, as well as an App and Connector Store for users to browse and download all of the UFDS or UFDS-related software. The final developed versions of these components will form UFDS Version 2, with the Living Labs receiving the needed support to proceed with the related software updates.

An illustration of this currently anticipated version is provided in Figure 3. Note that the specification of the architecture is an on-going process and updates will be provided in Deliverables D3.2 (UF Data Space Connector Store) and D3.4 (DISCO Data Space open software repository Ver 2, and Implementation Guide). Deliverable D3.4 in particular, will not only include a report on the final implemented architecture (UFDS Version 2) but also an envisioned architecture moving beyond the use cases described for the needs of DISCO and towards the needs of the urban freight sector and all its key players in general; this will be part of the documentation provided in D3.4 – DISCO Data Space open software repository Version 2 (M36).

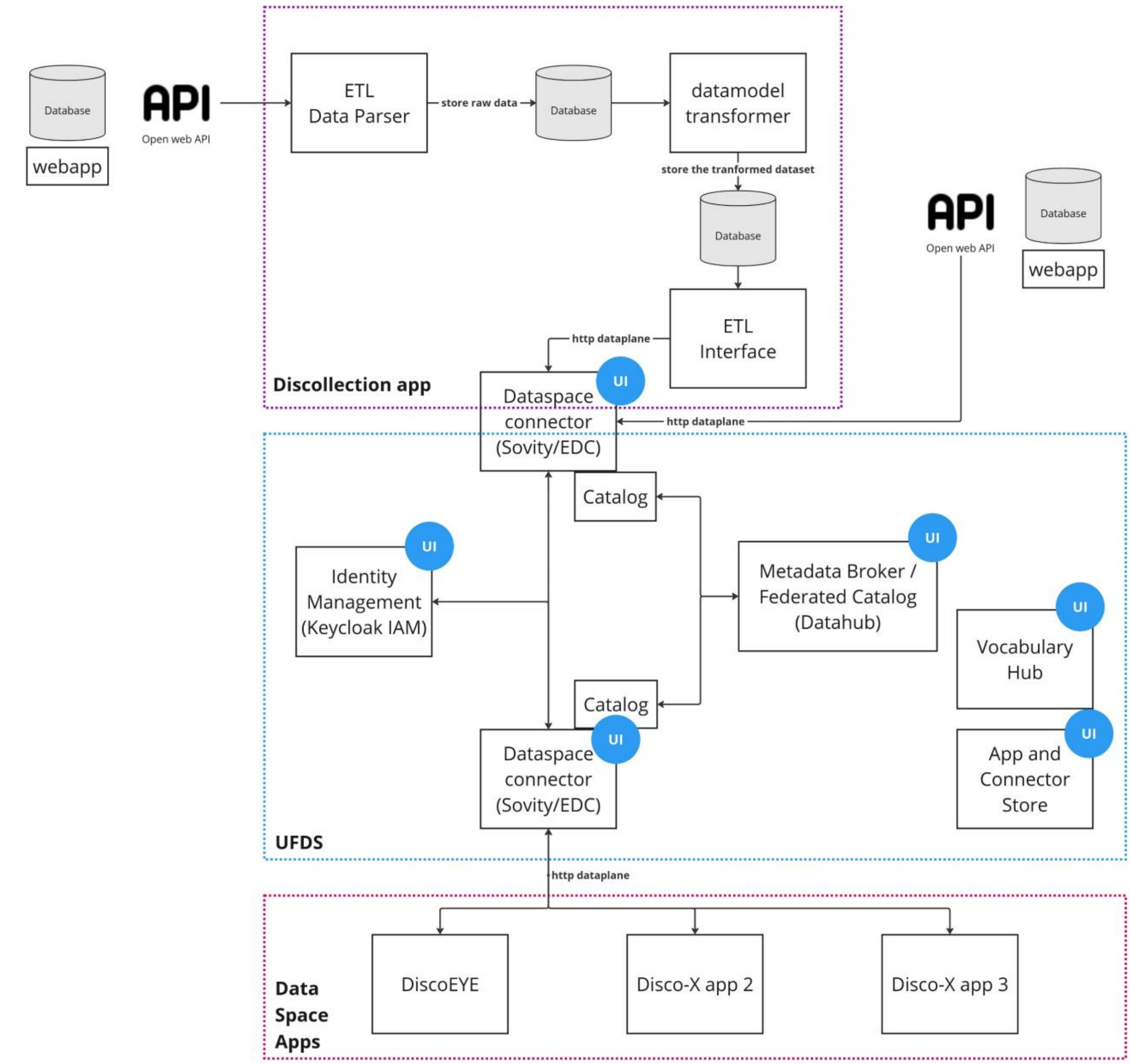

Figure 3: UFDS Version 2 high-level architecture (anticipated)

2.3. UFDS's data space connector and core components

Before proceeding with the per component description of the UFDS architecture, it should be noted that towards our architectural design, we have conducted extensive research into IDS-compliant or near-compliant FOSS solutions for all components, following our design principle (Use "general release" level mature technologies) stated in Section 2.2. For some components, such as the App and Connector Store (Section 2.3.42.3.4) and Vocabulary Hub (Section 2.3.5), compliant solutions were either unavailable or not well maintained. For these components, we plan to deliver components that embody the essential principles and functionalities outlined by the IDS reference architecture and although these components might not be IDS-compliant, they will adhere to the spirit and operational objectives of the IDS framework.

2.3.1. Data Space Connector

The Data Space Connector is the essential gateway or "key" to the data space, providing all participants, both data producers and consumers, with access to the Urban Freight Data Space (UFDS). To ensure robust and efficient data exchange, the connector must adhere to several architectural principles:

- *Simplicity and Efficiency:* The connector should be straightforward to use and maintain, featuring a small, efficient core with minimal external dependencies. This ensures that the connector remains lightweight and easy to manage, reducing potential technical debt and maintenance overhead.
- *Interoperability:* It should be platform-independent and capable of integrating seamlessly with diverse ecosystems. This interoperability ensures that participants using different systems and technologies can still effectively communicate and exchange data within the data space.
- *Decentralization:* The connector should support the deployment of necessary software components on the partners' side, ensuring data exchange occurs only between agreed points. This decentralized approach enhances security and control, allowing data to remain within the trusted boundaries of each participant's infrastructure.
- *Data Protection:* Only data linked to policies through contracts should be transferable, emphasizing data protection over sharing. This ensures that all data exchanges are governed by strict agreements, protecting sensitive information and complying with legal and regulatory requirements.
- *Separation of Metadata and Data:* Metadata should be handled separately from data to achieve high throughput rates during data transfer. This separation allows for efficient data handling and processing, reducing bottlenecks and improving overall system performance.
- *Consistent Semantics:* The connector should support data structured according to consistent semantics (ontologies), which are crucial for digital value creation. Consistent semantics ensure that data is interpreted correctly across different systems, facilitating meaningful data integration and analysis.

* *Automation:* All processes, from identity authentication to compliance with contractual data transmission and consumption regulations, should be automated where possible. Automation reduces the risk of human error, increases efficiency, and ensures consistent adherence to policies and agreements.
* *Standards Compliance:* The connector should adhere to the IDS standard and adopt the Dataspace protocol as far as possible. Compliance with established standards promotes interoperability, security, and reliability across the data space.

For UFDS version 1, Sovity's connector will be used. For UFDS version 2 and the further development and data space components additions required, we will consider both Sovity's and EDC's connectors; with Sovity's offering being based on EDC, not only their codebase but also their controlling APIs are similar, which will facilitate a switch from Sovity to EDC connector for UFDS Version 2 if required.

Both connectors offer a similar "management API"[2] that is well-documented[3] offering the following functionality:

* *HTTP Provisioner Webhook:* Called by an HTTP provisioner service to call when provisioning is complete. It sets up an endpoint for data retrieval in the consumer pull pattern.
* *EDR Cache:* Related to Endpoint Data Reference (EDR); dynamic endpoints with limited longevity for data pulling by the consumer. Provides additional metadata on a transfer process.
* *Catalog:* Allows querying the catalog of another connector, enabling the discovery of datasets and contract offers.
* *Contract Definition:* CRUD API for contract definitions, enabling creation, update, and deletion. Contracts link policies with assets through asset selectors and are only listable by the creator connector (i.e., the provider).
* *Contract Negotiations:* Initiates negotiations based on offers with counter parties. Also allows for querying negotiations, retrieving their state, and linking contract agreements. Useful for tracking unsuccessful negotiations.
* *Dataplane Selector:* Registers dataplanes for additional source/destination types, primarily relevant for remote dataplanes. For UFDS Version 1, the focus will be on the HTTP dataplane in order to connect with the REST API's offered by the Living Labs and DISCOLLECTION.
* *Policy Definition:* CRUD API for policies, allowing the creation, update, and deletion of policies. Policies are only manageable from the provider side.

---

[2] In fact, Sovity also offers the "Wrapper API" that is build on top of the management API. However, in an effort to not distance the codebase from the EDC connector, we chose to not use this API. We expect this to facilitate our switch from Sovity to EDC for UFDS Version 2.

[3] Sovity Swagger: https://github.com/sovity/edc-extensions/blob/main/docs/api/eclipse-edc-management-api.yaml
EDC Swagger: https://app.swaggerhub.com/apis/eclipse-edc-bot/management-api/0.6.0

- *Transfer Process:* Manages data transfers, including initiation, querying, pausing, resuming, terminating, and deprovisioning of resources. Both provider and consumer can track transfer states.
- *Transfer History:* Logs historical transfer data, detailing unique transfer process IDs and correlating contract and asset IDs between provider and consumer.
- *Asset:* CRUD API for assets, allowing the registration, update, and deletion of assets. Assets are only managed on the provider side.

2.3.2. Metadata Broker / Federated Catalog

The metadata broker, also known as the federated catalog, serves as a repository that stores and manages metadata from various data assets within the data space's connectors. By maintaining this federated catalog, the metadata broker enables data space participants to browse and identify relevant datasets they require, along with the connectors offering these datasets.

Each connector in the data space announces its existence to the broker or alternatively, the broker is provided with a list of participating connectors on the data space, in this latter case this is part of the onboarding procedure of new participants. Following this, the broker queries the catalog API of each connector at regular intervals. This API, as mentioned in Section 2.3.1, allows connectors to share their data assets' metadata with each other. The broker collects this information, storing all the metadata concerning the datasets and the connectors offering them.

The federated catalog maintained by the broker is accessible to all data space participants. This accessibility allows participants to search for and locate the data relevant to their needs, facilitating effective data utilization within their living labs. Thus, the broker ensures a cohesive and comprehensive view of the available data assets, promoting efficient data discovery and integration across the data space.

The EDC Connector and Sovity's platform take a different approach with regards to implementing the Broker functionality. Specifically, in the EDC universe, this functionality materializes via the federated catalog, a plugin that can be added to all EDC connectors. In this way, each connector can deploy its own "personal" Broker that is then tasked with polling all known connectors in the data space and keep a record of their data assets. On the other hand, Sovity's platform offers a broker component, which will be evaluated for inclusion in UFDS Version 1 and/or later versions. This component, although also tasked with polling all connectors known to it for updated on their data asset offerings, like the EDC connector, is independent from the Sovity connector and can be accessed by all connectors in the data space, in contrast to the EDC connector.

An alternative option for accessing the functionality of a metadata broker is through a DataHub-backed federated catalog. DataHub, an open-source metadata platform designed for the management and discovery of data assets across various systems, offers an intuitive UI that enables users to catalog different datasets. It also provides robust features for searching, discovering, and controlling access to these datasets.

Figure 4 illustrates the interactions between users, a DataHub-backed federated catalog, and the rest of the data space's connectors.

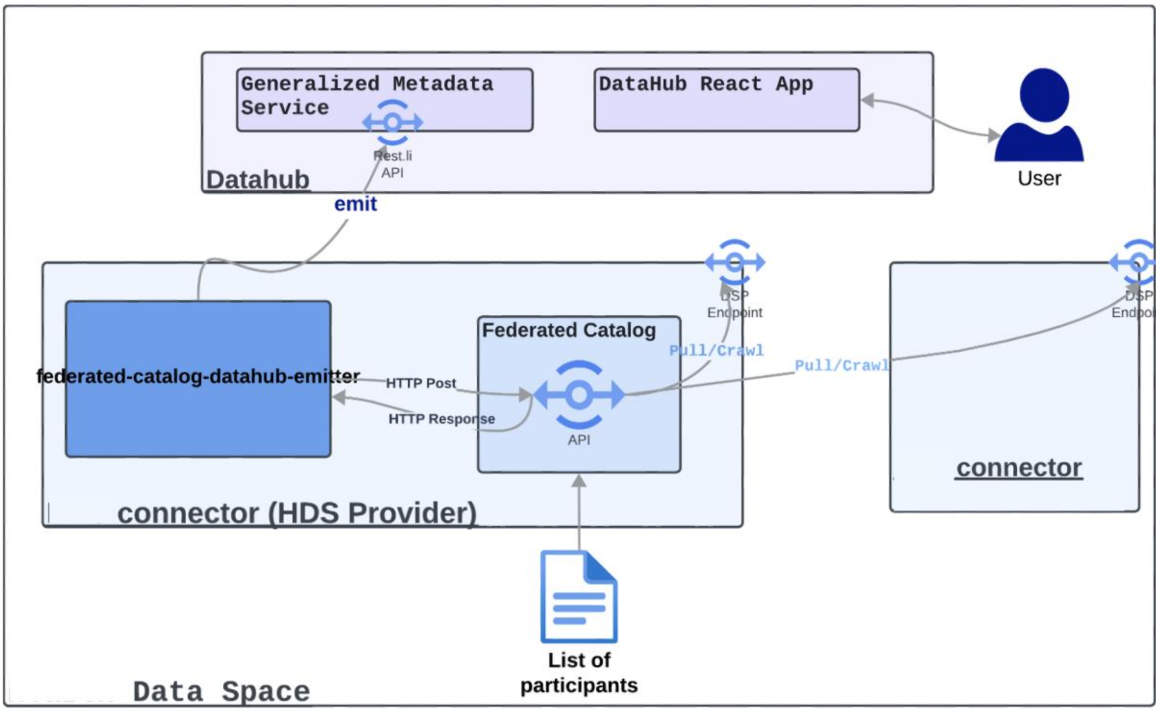

Figure 4: Interaction between Users, DataHub Metadata Broker, and Connectors.

2.3.3. Identity Management

With a Data Space having many participants interacting, robust means of identifying those participants are required. The IDS architecture provides mechanisms that aid participants in establishing this required level of trust when interacting. This is achieved by establishing trusted connections for every connector-to-connector interaction. The identity management framework comprises three complementary components: Certificate Authorities (CAs), Dynamic Attribute Provisioning Service (DAPS), and a Participant Information Service (ParIS). Specifically:

- **Certificate Authorities (CAs):** Responsible for issuing identification certificates (keys) for each connector-participant in the data space. CAs are envisioned to be controlled by trust building entities that can credibly ensure only registered participants can operate in a particular data space.
- **Dynamic Attribute Provisioning Service (DAPS):** Verifies the status and validity of the CA issued keys and authorizes connector-to-connector interactions by ensuring that all participating connectors are correctly identified.

- **Participant Information Service (ParIS):** Provides business-related information about each connector-participant after conducting the necessary background checks. It functions similarly to a metadata broker but focuses on participants rather than datasets.

Figure 5, sourced from IDSA's RAM Version 4, illustrates an interaction between IDS Connectors and Identity Components.

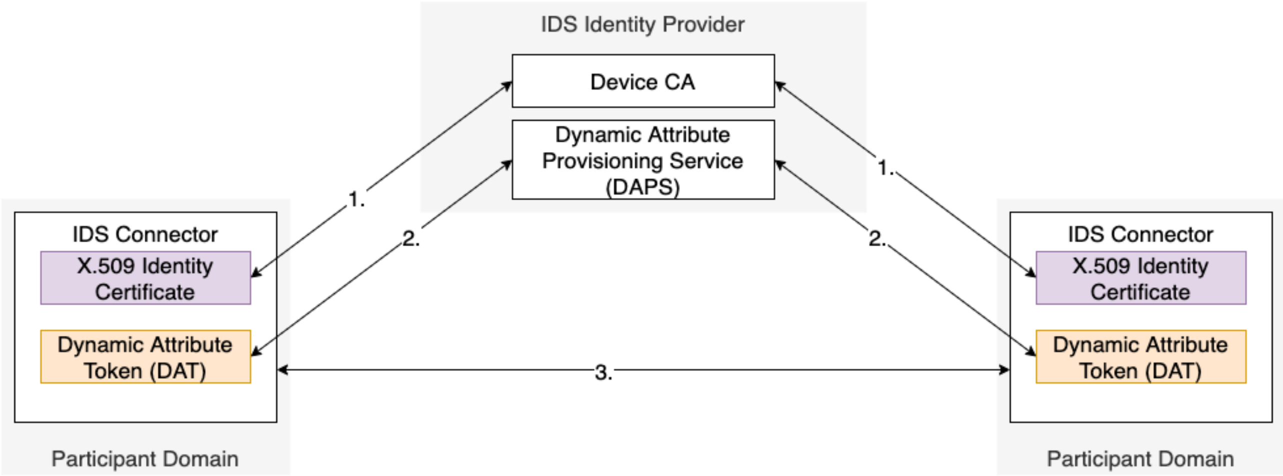

Figure 5: Interaction between IDS Connectors and Identity Components (source: IDSA Reference Architecture Version 4)

In the DISCO project, considering the needs and research nature of the project, we have decided to develop a DAPS component within the UFDS architecture, incorporate a PoC solution for the CA component, and document the usage and benefits of the ParIS component in the documentation provided as part of D3.4 – DISCO Data Space open software repository Version 2 (M36), for more information on this see the implementation's Next Steps (Section 3.2). This approach ensures that all participants in the UFDS undergo a comprehensive and documented validation process prior to being allowed to interact with other connectors on the data space. Participants will create and send their public key to the CA, which will then be added to the DAPS. Due to the research nature of the DISCO project, the CA authority will be managed by INLE and AKKA, the technical developers of the UFDS and administrators of the Open Software Repository (Section 4).

From a technical standpoint and based on operational Data Space implementations that we have witnessed[4,5,6], the CA is not a technical component but rather a documented manual process. This process included background checks on new participants, after which their credentials are added to the DAPS to enable secure communication within the data space. Concerning the DAPS component,

[4] Health Dataspace (https://www.imec.be/nl/vlaamse-innovatiemotor/impactdomeinen/smart-health)
[5] Citcom.ai (https://citcom.ai/)
[6] Boot-X Dataspace (https://www.boot-x.eu/)

our research identified two FOSS candidates; EDC's Identity Hub[7] and Sovity's DAPS[8]. With both EDC and Sovity's solutions implementing their identity provider components via the Keycloak IAM platform.[9]

Keycloak is an open-source identity and access management (IAM) solution designed to provide robust identity management and single sign-on (SSO) capabilities for modern applications and services. It supports standard protocols like OAuth 2.0, OpenID Connect (OIDC), and SAML 2.0, allowing seamless integration with diverse systems. Keycloak features user federation with LDAP and Active Directory, social login integration, fine-grained authorization, and extensive customization options. Its admin console enables easy management of users, roles, and clients, while the user console offers self-service account management.

Architecturally, Keycloak is designed to be modular and scalable, supporting standalone and clustered deployments. Key components include realms (security domains), clients (applications requesting authentication), users and groups, and roles. The system prioritizes security with multi-factor authentication (MFA), encrypted communications, and detailed audit logging. Its extensibility allows for custom authentication flows and user storage providers, making it a versatile choice for securing applications and services across various environments.

Technical flow

The Data Space Connector employs an OAuth2Service extension for identity services using the OAuth2 protocol. This extension consists of two libraries: oauth2-client and oauth2-core. While the client library supports both the Client Credentials Grant flow with client id/secret and private key JWT, the OAuth2Service specifically uses private key JWT for secure authentication. This flow works as follows:

1. *Key Pair Setup:* The connector holds a public-private key pair stored in a PKCS12 (pfx) file. This key pair is crucial for multiple operations, including the secure exchange of data in the consumer-pull data flow. The private key is used to sign authentication requests, while the public key is shared with the identity provider (IdP) for verification purposes. The public key is made available to the identity provider (this can be a simple copy-paste or it can be extracted from the .crt or via a JWKS endpoint).
2. *Public Key Sharing:* The public key from the connector's key pair must be shared with the IdP. This can be achieved in several ways:

[7] https://github.com/eclipse-edc/IdentityHub
[8] https://github.com/sovity/sovity-daps
[9] https://www.keycloak.org/

- Copy-Paste Method: Manually copying the public key and providing it to the IdP.
- Certificate Extraction: Extracting the public key from a certificate (.crt) file.
- JWKS Endpoint: Utilizing a JWKS (JSON Web Key Set) endpoint where the public key is published and can be accessed by the IdP. The IdP verifies this JWT and issues an access token that can then be used by the connector.

3. *Token Request:* To obtain an access token, the connector sends a request to the IdP. This request includes a JWT (JSON Web Token) that has been signed with the connector's private key. The JWT includes claims (information) about the connector, ensuring that it can be authenticated securely. Upon receiving the request, the IdP verifies the signed JWT using the public key that was previously shared.
4. *Token Issuance:* If the JWT in the precious step is valid, the IdP issues an access token. This token serves as a proof of authentication and authorization for the connector to access specific resources.
5. *Token Verification:* The access token issued by the IdP can be verified by any service or application using the IdP's public key. This is facilitated through the JWKS endpoint, where the public key is published. Verification ensures that the token is legitimate and has been issued by a trusted IdP, thus maintaining secure interactions within the data space.

The above discussed technical flow is illustrated below in Figure 6 (steps 1 through 3, low-level) and Figure 7 (steps 1 through 5, high-level).

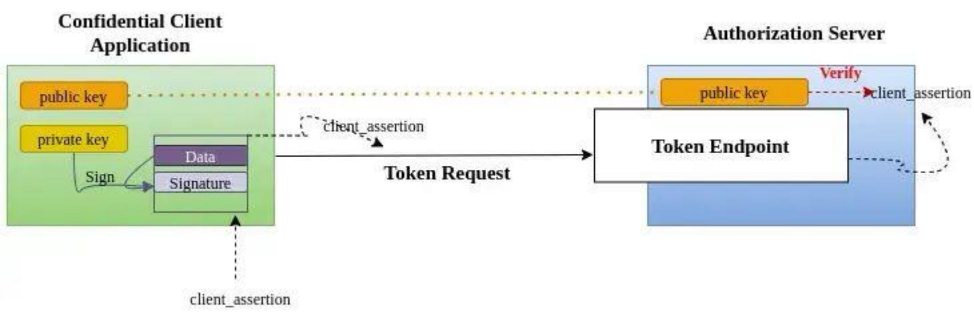

Figure 6: Idenity Management Technical Flow - Steps 1-3

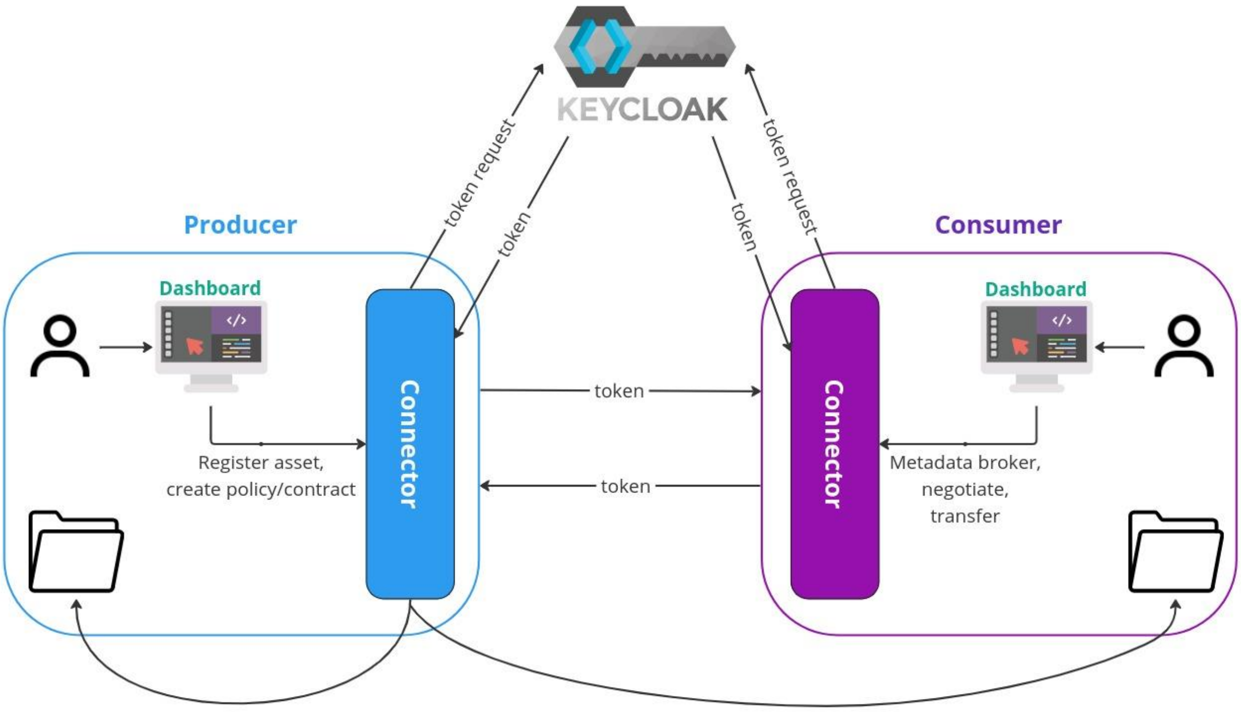

Figure 7: Identity Management Technical Flow – Steps 1-5 (high-level). Integration of Keycloak in the architecture.

2.3.4. App and Connector Store

As previously explained, the DISCOEYE application will be developed for UFDS Version 1, while support for further applications will be a focal point of the architecture and development plan for UFDS Version 2. All developed applications, their accompanying connectors, as well as all other components of the UFDS will be accessible by the data space participants through the App and Connector Store component.

While IDS-RAM Version 4[10] outlines the architecture of the IDS App Store, there appears to be a lack of recent development progress in this area. Neither EDC nor Sovity currently offer such a component, and there have been no significant development efforts on the IDS's offering[11] since September 2021. Moreover, IDSA's latest Data Connector Report[12] does not include any further

---

[10] https://docs.internationaldataspaces.org/ids-knowledgebase/v/ids-ram-4
[11] https://github.com/International-Data-Spaces-Association/IDS-AppStore
[12] https://internationaldataspaces.org/wp-content/uploads/dlm_uploads/IDSA-Data-Connector-Report-No-15-May-2024.pdf

information for an App Store, or for App Store support by any of the reviewed Data Space Connectors.

Given the circumstances, we have decided to develop our own version of an App and Connector Store. This store will align with the spirit of IDSA's reference architecture but is not currently projected to be IDS compliant. Nonetheless, we plan to create website allowing users to search and filter through UFDS applications and connectors and be able to download and deploy them on their machines. Assuming the UFDS registration process has been already completed on their side, this should be sufficient for them to connect and use the data space.

From a technical standpoint, the process is presented below (subject to minor deviations during development). It assumes that the user does not have a prior installation of UFDS's "core applications" on their machine which is required to be installed first (steps 1-3) prior to the installation of any other application (steps 4-5). A more user-centric and non-technical perspective of this process is presented in Section 4.1.

1. *Core Applications Install Script Download:* The user accesses the App Store and downloads the "core applications install script." This script includes the connector and all necessary components for connecting, finding, and exchanging information on UFDS at a base level.
2. *Preparation of Installation Environment:* The user places the install script (which is OS-dependent) inside an empty directory alongside their private key. The corresponding public key would have been previously used to register the user or organization with the UFDS.
3. *Execution of Core Installation Script:* The user runs the install script. This action triggers the following automated sequence:
    - The script downloads the required files from an open software repository.
    - It installs and deploys the connector, utilizing the provided private key.
    - All necessary components are installed on a specific Docker network on the user's machine, ensuring a contained and consistent environment.
4. *Application Download:* The user selects the desired application from the App Store and downloads its corresponding script.
5. *Application Deployment:* The user places the downloaded application script inside the same directory as the core installation script and runs it. This triggers another automated sequence:
    - The script downloads the necessary application files from the open software repository.
    - It deploys the application on the same Docker network, ensuring seamless integration with the existing data space components.

Below, we provide an overview of DISCOEYE functionality, UFDS's first application.

DISCOEYE application

DISCOEYE, a Geo-Data Visualization Tool is a pivotal application designed for the UFDS data space, enabling users to interact and visualise geo-located datasets in an intuitive and visually engaging manner. At its core, this application allows users to filter through a variety of data sources based on the connector offering the data, the datasets themselves, and/or by using labels that can be provided to describe different datasets offered by each connector on the data space.

All datasets integrated within this application contain geo-located data points, which are visually represented on an interactive map based on mapbox software libraries.[13] Users can seamlessly navigate the map, zoom into specific regions, and view various geographical elements. Specifically, each item on the map corresponds to a specific location (one data point) or zone (a collection of related data points). When a user clicks on these points or polygons, a pop-up window appears presenting the user with the properties and metadata of the selected location or zone. This includes attributes such as coordinates, data source, and other relevant information, facilitating a deeper understanding of the spatial data. DISCOEYE thus transforms complex geo-datasets into accessible and actionable insights, acting as the basis of data-driven decision-making within the UFDS ecosystem.

2.3.5. Vocabulary Hub

To enhance seamless interoperability among various data space applications, components, and datasets, it is essential to establish a set of commonly recognized, standardized terms to describe the datasets both at a high-level and granular level. This set of agreed-upon terms forms a vocabulary, which must be shared among all data space participants. Additionally, this vocabulary must be machine-readable, with its descriptions and titles being queryable to a certain extent, as specified by the IDS reference architecture.

In the DISCO project, living labs are distributed across different countries. While interoperability issues are not expected to arise in UFDS version 1 due to the smaller number of datasets utilized, it is anticipated that in UFDS's future versions, different interpretations of datasets and underlying data points will need to be addressed.

To ensure that datasets remain interoperable, we will aim to take advantage of ontologies to create metadata elements to annotate and describe these datasets. This approach will make the various datasets usable across the different living labs of the project. To this end, we have identified the work by the Alliance for Parking Data Standards (APDS) on the APDS ontology as a suitable candidate

[13] Note that the usage of these libraries when handling geo-located data sourced via different connectors over the http dataplane can form the basis of other applications which will utilize geo-located data as input and/or output.

for addressing the needs of a significant portion of the datasets that the UFDS will handle. A detailed discussion of this ontology is presented in Appendix: APDS Ontology.

Furthermore, TNO's implementation of the Vocabulary Hub, known as the Semantic Treehouse[14], offered under an Apache 2.0 license (i.e., FOSS), has been identified as an appropriate solution for meeting the aforementioned requirements of a data space. It provides a platform that allows users to host, maintain, publish, and document relevant vocabularies effectively.

2.4. DISCOLLECTION

The DISCOLLECTION application is a middleware that prepares the data from the living labs to be onboarded on the data space. It follows the principles of an ETL data pipeline (Extract, Transform, Load). The data is extracted from the Living lab API's and transformed into a common data model. The transformed data is then stored to create a historical data footprint. Following this, and depending on the data space connector utilised, a different load procedure is followed. Specifically, in the case of the Sovity connector, it is loaded on it via an HTTP dataplane fully utilizing the REST API query capabilities of the ETL data pipeline. On the other hand, in the case of the EDC connector, the ETL pipeline will use the same tools until the EDC connector. For the EDC connector, DISCOLLECTION will need to change the target endpoint from Sovity to EDC. This update will be minor and will take place when and if we are required to switch connectors.

DISCOLLECTION will take the role as a data intermediary outside of the data space. It will gather data from the Living Labs and transform them when ingested towards the chosen data model. For UFDS Version 1, and as mentioned above, the DISCOLLECTION app will be connected via HTTP dataplane to the Sovity connector and become a data provider to the data space. DISCOLLECTION will serve as the **Smart Data Platform (SDP)** for each Living Lab. In the architecture shown in Figure 8 the LL will provide an easy to use API that can be used to retrieve data over the web. For UFDS1 the DISCOLLECTION app will have to be installed in at least one Living Lab. The UFDS version 1 will have the DISCOLLECTION application already fully implemented for the Gent data, meaning that the ETL data pipeline is already collecting data from the Living labs for a few months. The data following the ETL pipeline will use the extract, transform and load pipeline. The data is extracted from the Living lab REST API's and stored in the ETL database. The data is then transformed into a common data model standard. The ETL pipeline connects with the Sovity connector through a REST API and is able to query the database. Needed features will be added later on in the data space development cycle (i.e., past UFDS Version 1). The DISCOLLECTION application is going to use a mix of commonly used ETL tools together with a custom interface to connect to the Sovity connector. The tools used

[14] https://www.semantic-treehouse.nl/

and alternatives options will be documented in the DISCOLLECTION manual which will be documented in Deliverable D2.4 – DISCO-X Enabling Tools (M18).

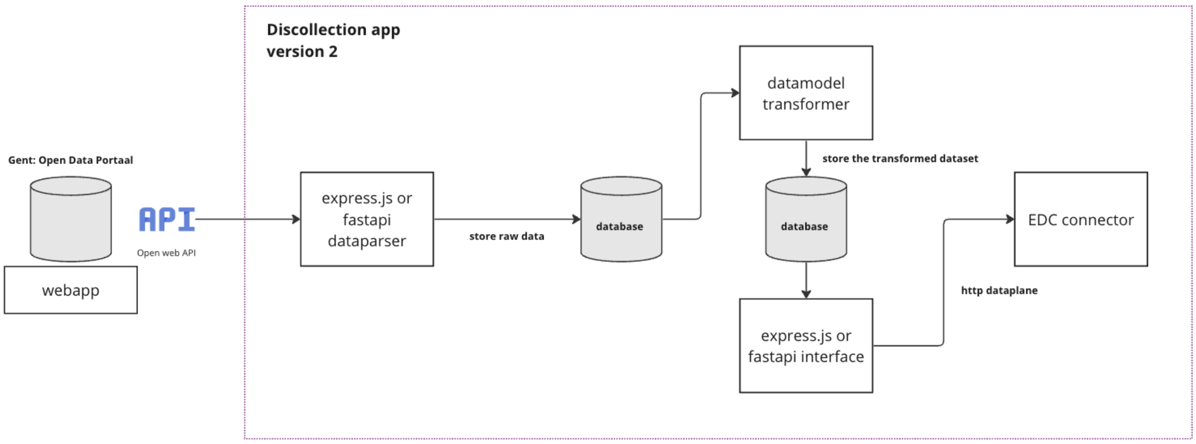

Figure 8: Discollection application that extract, loads and transforms the data

DISCOLLECTION will evolve after the first PoC (UFDS Version 1) into a global solution for all the Living Labs with the Living Labs have the possibility to onboard the data they are going to work around through the DISCOLLECTION app. Towards UFDS version 1, the DISCOLLECTION application will work with the Gent data Portal to transform their parking data to be offered (see Section 3.1.1) in the APDS format (see Appendix: APDS Ontology). This data will be stored and transformed in the application and be made ready for the Sovity connector to be transferred to the DISCOEYE application (see Section 2.3.4).

3. Implementation

As briefly explained in Section 0, the development of the Urban Freight Data Space (UFDS) will deliver a first version of the data space, UFDS Version 1. In Section 3.1 below we detail the data owners, data involved, and use cases that this version will cover. Following this, in Section 3.1.2 we discuss the processes we intend to follow in order to support further use cases and the datasets these will involve, by a second technical version of the data space, UFDS Version 2, delivered in Q4 2025. Moreover, this section also considers the process that will result in use cases that will not be technically supported by UDFS Version 2, due to the technical shortcomings discussed in Section 0, but the feasibility to cover these in future (post-DISCO) UFDS extensions will be documented.

3.1. Urban Freight Data Space - Version 1

Towards UFDS Version 1 and our need to assess the types of datasets that could be offered on the data space, and the kinds of use cases that it could cover, we held several meetings with most of the Starring Living Labs' participants in December 2023. Later, in early February 2024, we conducted an online workshop[15], followed by additional meetings with participants from both the Starring and Twinning Living Labs on this matter.

In detail, the purpose of these meetings and the workshop was to gain insights into the needs, requirements, and expectations of the living labs from the UFDS, inquire about the datasets that could be shared for the needs of the UFDS, and, using this information, draft UFDS use cases feasible for demonstration in DISCO. Another outcome of this process was to begin identifying the architectural requirements these use cases would entail for the UFDS.

The outcome of this engagement was the receipt of some dataset information from the Thessaloniki, Ghent, and Helsinki Starring Living Labs, while the Copenhagen Living Lab has agreed to provide us with relevant data by Q3-Q4 2024. Furthermore, we engaged in discussions with all of the Twinning Living Labs on their data needs and possible dataset offerings. In addition, the datasets provided by Ghent, Thessaloniki, and Helsinki are accessed through various RESTful API services, as is typical in these cases, and include documentation covering their consumption. However, it should be noted that the provided datasets did not fully cover the data needs of the DISCO-X applications; we further discuss this in Section 3.2 below.

Before proceeding to detail the use cases covered by UFDS Version 1, we first present below the datasets that will be available in this version (and any future versions) of the data space.

3.1.1. Data owners and datasets provided

Ghent

Ghent allowed us to make use of the datasets available on its "Ghent Open Data Portal"[16] illustrated in Figure 9. After reviewing the datasets available, we identified the following two datasets as fitting for the demonstration purposes of UFDS Version 1. It should be noted that further fitting datasets exists, and these will be offered later on during the data space development process.

[15] Workshop's title: Workshop on Urban Freight Data Space and input required from the Living Labs
[16] https://data.stad.gent/explore/?disjunctive.keyword&disjunctive.theme&sort=modified

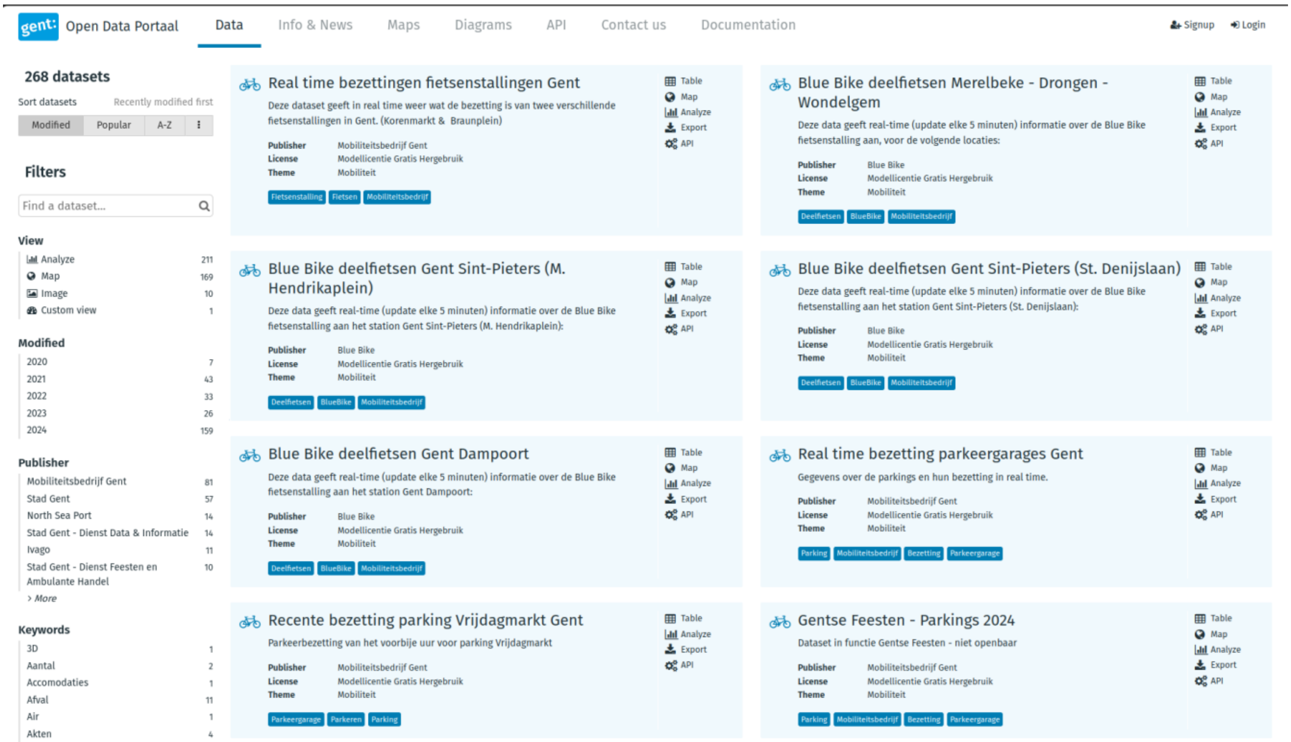

Figure 9: Ghent's Open Data Portal

1. **Public parking spaces and real time occupation:**[17] This dataset describes the occupation of the parking spaces in the public parking garages in Ghent. The data is provided in real-time and is updated in 5-minute intervals. It includes information for 13 different garages around the city of Ghent, covering among other things their location, maximum capacity, current capacity, and opening times. The information provided for one of these garages is illustrated in Figure 10.
2. **Street parking locations:**[18] This dataset describes the geospatial locations of the curbside parking space in Gent. Each datapoint is the location of each parking space that can fit one or more vehicles, depending on the size of the vehicles. The dataset is static and does not provide occupancy data; it is scheduled to be updated on a yearly basis. An example data-point of one such parking zone is provided by Figure 11.

[17] https://data.stad.gent/explore/dataset/bezetting-parkeergarages-real-time/table/?sort=-occupation
[18] https://data.stad.gent/explore/dataset/straatparkeerplaatsen-gent/api/

```
{
    "name": "Savaanstraat",
    "lastupdate": "2024-06-25T16:27:52+02:00",
    "totalcapacity": 510,
    "availablecapacity": 287,
    "occupation": 43,
    "type": "carPark",
    "description": "Ondergrondse parkeergarage Savaanstraat in Gent",
    "id": "https://stad.gent/nl/mobiliteit-openbare-werken/parkeren/parkings-gent/parking-savaanstraat",
    "openingtimesdescription": "24/7",
    "isopennow": 1,
    "temporaryclosed": 0,
    "operatorinformation": "Mobiliteitsbedrijf Gent",
    "freeparking": 0,
    "urllinkaddress": "https://stad.gent/nl/mobiliteit-openbare-werken/parkeren/parkings-gent/parking-savaanstraat",
    "occupancytrend": "unknown",
    "locationanddimension": "{\"specificAccessInformation\": [\"inrit\"], \"level\": \"0\", \"roadNumber\": \"?\", \"roadName\": \"Savaanstraat 13\\n9000 Gent\",
      \"contactDetailsTelephoneNumber\": \"Tel.: 09 266 29 40\", \"coordinatesForDisplay\": {\"latitude\": 51.04877362543108, \"longitude\": 3.7234627726667133}}",
    "location": {
        "lon": 3.7234627726667133,
        "lat": 51.04877362543108
    },
    "text": null,
    "categorie": "parking in LEZ",
    "dashboard": "True"
}
```


Figure 10: Ghent's public parking and real time occupation dataset, data-point example.

```json
{
    "geometry": {
        "type": "Feature",
        "geometry": {
            "coordinates": [
                [
                    [
                        3.723090419715097,
                        51.06195288721567
                    ],
                    [
                        3.723272348323068,
                        51.06191392940891
                    ],
                    [
                        3.72328280650961,
                        51.061930682557055
                    ],
                    [
                        3.723098630316802,
                        51.06197012080112
                    ],
                    [
                        3.723090419715097,
                        51.06195288721567
                    ]
                ]
            ],
            "type": "Polygon"
        },
        "properties": {}
    },
    "pregime": "1",
    "opmerkingen": null,
    "capaciteit": 2,
    "datum": "2019-01-09T23:00:00Z",
    "straatcode": 70232,
    "pcapcode": 3103,
    "wijkcode": null,
    "ptype": 100,
    "zone": "TARROOD",
    "geo_point_2d": {
        "lon": 3.723186061448406,
        "lat": 51.06194192223374
    }
}
```

Figure 11: Ghent's street parking locations dataset, data-point example.

Thessaloniki

The datasets provided by Thessaloniki are part of the Thessaloniki Urban Logistics Data space which is fully documented[19] as shown in Figure 15. Specifically, the datasets available to us are the following:

1. **Shops:** A list of food and beverage establishments in the municipality of Thessaloniki. The list currently includes 1600+ establishments separated in three broad categories, (i) Super Markets, (ii) Restaurant / Cafe / Bar / Entertainment centre / Hotel unit, and (iii) Mini

[19] REST API Documentation: https://uldthess-api.imet.gr/docs#/

Markets. The dataset contains location information per establishment. An example of is information can be found in Figure 13.

2. **Warehouses:** A list of warehouses in the municipality of Thessaloniki. The list currently includes 3 warehouse owners -and a few anonymous owners- with a total of 92 warehouse compartments. The dataset contains location and storage capacity information per warehouse. An example of is information can be found in Figure 12.

3. **Lockers:** A list of lockers to send/receive mail and small parcels. The list currently includes about 500 lockers belonging to one of three delivery companies. The dataset contains location information per locker. An example of is information can be found in Figure 14.


```json
{
  "business_activity": null,
  "date_created": "2024-04-04T06:34:40",
  "date_edited": "2024-04-04T06:34:40",
  "description": "Εκθεσιακό Περίπτερο",
  "name": "Περίπτερο 6",
  "owner": "tif",
  "type": "real_estate",
  "warehouse_id": 6,
  "website": "www.helexpo.gr",
  "x_coordinate": 40.62567,
  "y_coordinate": 22.95619,
  "compartments": [
    {
      "capacity_area": 2805,
      "capacity_pallets": 0,
      "capacity_type": "area",
      "compartment_id": 15,
      "date_created": "2024-04-04T06:34:40",
      "date_edited": "2024-04-04T06:34:40",
      "max_pallet_height": 0,
      "max_pallet_weight": 0,
      "name": "Περίπτερο 6"
    },
    {
      "capacity_area": 840,
      "capacity_pallets": 0,
      "capacity_type": "area",
      "compartment_id": 16,
      "date_created": "2024-04-04T06:34:40",
      "date_edited": "2024-04-04T06:34:40",
      "max_pallet_height": 0,
      "max_pallet_weight": 0,
      "name": "Περίπτερο 6 -Δώμα"
    }
  ]
}
```


Figure 12: Warehouse datapoint example.


```json
{
  "daily_loading_unloading_frequency": 4,
  "geometry": {
    "x": 40.603294779386296,
    "y": 22.96775285680377
  },
  "shop_category": "Restaurant/Cafe/Bar/Entertainment centre/Hotel unit",
  "attributes": {
    "ADDRESS": "25ης Μαρτίου  127 Θεσσαλονίκη",
    "ANTIKEIMENO": "ΑΝΑΨΥΚΤΗΡΙΟ",
    "KODIKOS_KATASTHMATOS": "28308",
    "OBJECTID": 18,
    "city": "Θεσσαλονίκη",
    "dimotiki_koinotita": "Ε' ΔΗΜΟΤΙΚΗ ΚΟΙΝΟΤΗΤΑ",
    "number": 127,
    "street": "25ης Μαρτίου ",
    "x": 412517.1098,
    "y": 4494946.3792,
    "zip": 54249
  }
}
```


Figure 13: Shop datapoint example.


```json
{
  "description": "Φαρμακείο Κυριακίδου (Ασβεστοχώρι)",
  "owner_title": "ΣΚΡΟΥΤΖ ΥΠΗΡΕΣΙΕΣ ΔΙΑΝΟΜΗΣ",
  "x": 40.64,
  "y": 23.029
}
```


Figure 14: Locker datapoint example.

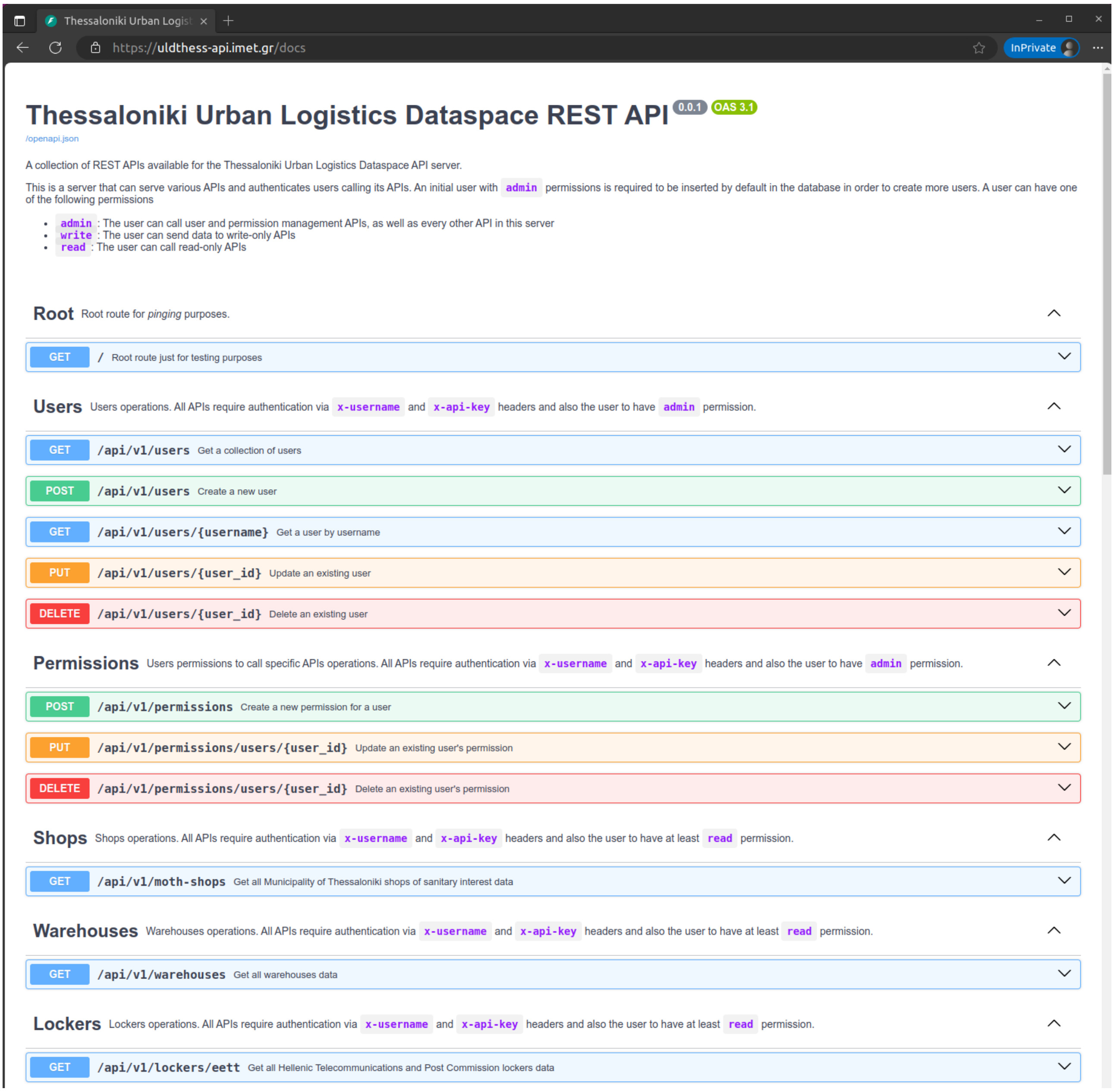

Figure 15: Thessaloniki Urban Logistics Data Space - REST API documentation.

3.1.2. Use Cases

For UFDS Version 1, our primary aim is to set up a stable and resilient software foundation that will provide the basic communication functionality required for a data space, in the sense of a minimum viable data space. This will include the initial versions of the UFDS connector and the necessary components and functionality for a participant-connector to (i) register on UFDS, (ii) find other connectors, and (iii) produce and consume data within the data space. The architecture of this first version can be seen in Figure 2 above.

Furthermore, since most datasets related to urban freight and related services will directly or indirectly involve geography, whether in the underlying data as is the case for all the datasets to be offered in version 1 (detailed in Section 3.1.1) or in the results of services using this data (e.g., a map routing tool), we believe that the needs of applications on the UFDS will revolve around models that have access to a geography-enabled UI.

For this reason, and since the UFDS use cases are planned to be fully defined later in the year, we have decided that a foundational use case for the UFDS will be to map the available datasets on a map-type application. Users will be able to filter through different datasets, visualize them on the map using the data's geospatial information, and click on them to gain further insights into each datapoint.

This will be achieved through the development of an initial data space application, DISCOEYE, which will offer intuitive and user-friendly visualization of the geospatial-related datasets available to the data space participants. Additionally, as explained in Section 2.3.4, it will provide a software foundation based on the mapbox software libraries, offering essentially a building block and source-code foundation for the majority of future applications to be developed on UFDS.

The users of UFDS version 1 will be those interested in using DISCOEYE which provides a comprehensive and intuitive visualization of the datasets available on UFDS, allowing users to easily explore and understand the breadth of data on offer. Targeted user groups include (i) prospective application developers and (ii) Living Lab stakeholders for to the following reasons.

First, DISCOEYE will enable application developers to visualize the data in a manner that allows them to gain insights into the potential applications and services that can be built using the data. This can inspire novel uses and innovative solutions, enhancing the overall utility and impact of the UFDS.

Second, for Living Lab stakeholders who play a crucial role in the ecosystem, DISCOEYE will allow them to view the data firsthand and better articulate their needs and requirements from the UFDS. This improved understanding will facilitate a more effective and faster drafting of further use cases, ensuring that the data space evolves in a way that meets the practical needs of its users.

3.2. Next Steps

Following UFDS Version 1, several key steps will guide our efforts to further develop and refine the data space.

Specifically, with the upcoming finalisation of the UFDS use cases in the coming months (by Q4 2024), there are still further datasets to become available to UFDS including possibly information on traffic, parking and related pricing, city zoning information (e.g., low emissions zones), planned events (e.g., concerts), LSP demand data, and logistics real estate.

It is in our immediate plan to study the exact data requirements of the various DISCO-X measures and their scope of innovation and based on (i) the already offered datasets, (ii) the anticipated datasets, and (iii) the "spirit" of the DISCO-Xs, develop blueprints for data space applications that effectively utilize this data. Then, we plan to consult our partners in WP2 and WP4 on these blueprints; should any additional datasets be required, we will request their provision from the Living Labs or seek guidance on how to construct or replicate them. Table 1, illustrates the data types required by each DISCO-X; note that although these requirements are derived from D2.1 Urban Logistics Transition Requirements, the various data types shown there have been consolidated here in five broader categories.

As indicated in Section 3.1, we have already received datasets that fall within the categories of parking (Ghent) and city/warehouse data (Thessaloniki). Additionally, we anticipate receiving further datasets pertaining to parking from Copenhagen in the next months, as well as further city-type datasets from all starring Living Labs as well as some traffic-related datasets. On the other hand, it is unlikely that all the required datasets will become available for the project. Reasons for this include cities' inability to provide the data in time (e.g., digital versions of city zoning information do not yet exist) and some stakeholders' reluctance to share critical operational data (e.g., LSP-related datasets) for the reasons stated in Section 0. Additionally, the datasets provided so far do not overlap completely; for instance, we have parking data but no logistics real estate information from Ghent, while in Thessaloniki the situation is reversed. To address this, as mentioned previously, we will synthesize some of these datasets. This approach involves gathering information on the expected structure of the required datasets from relevant stakeholders and/or using data from one city (e.g., parking data from Ghent) to fabricate equivalent datasets for another city (e.g., Thessaloniki). The objective is to create a comprehensive dataset for at least one city, which will allow us to demonstrate the UFDS use cases.

Another goal of our next steps involves data standardization which we will focus on via two components of the UFDS architecture, DISCOLLECTION (see Section **Error! Reference source not found.**) and the Vocabulary Hub (see Section 2.3.5). In this context, we have identified the APDS ontology, which is presented in Appendix: APDS Ontology. Our goal here includes restructuring some datasets as a proof of concept using this ontology and, in addition, incorporating it into the data space application blueprints. This will ensure that datasets expressed using the APDS ontology will be compatible with the data space applications developed.

Table 1: Summary of DISCO-Xs data requirements (based on D2.1)

	DISCOCURB	DISCOPROXI	DISCOESTATE	DISCOBAY
Parking Data	☑			

City Data	☑	☑		
Traffic Data	☑	☑		
Real Estate Data		☑	☑	☑
LSP Data		☑	☑	☑

The above process will result in the second version of the data space, UFDS Version 2, which will be delivered in Q4 2025.

Moreover, the final architecture (see Figure 3 for an anticipated version) will not support all use cases. For this reason we also plan to document the feasibility of covering these in the future, originating from UFDS Version 2 architecture, as well as ensuring the continuous improvement of the UFDS by recording our experience gained in the DISCO project during the development of the first two versions. This documentation will act as a valuable resource for future developers and stakeholders involved in the urban freight data space and will include a listing of best practices, technical hurdles encountered, and innovative solutions implemented to overcome said challenges, as well as a roadmap for potential future projects and research initiatives, inspired by the needs, requirements, and trends observed during the DISCO project.

The next two technical deliverables of WP3 will provide relevant updates and reports on the above. Specifically, D3.2 – UF Data Space Connector Store (M24) will provide a report on the implementation of UFDS Version 1, and an update on these next steps and procedures. Similarly, D3.4 – DISCO Data Space open software repository Version 2 (M36) and Implementation Guide will provide a report on UFDS Version 2 as well as the documentation mentioned above.

4. Open Software Repository

4.1. Description

The purpose of the open software repository is to serve as a technical platform where UFDS components, or UFDS-related components developed during the project, can be accessed, shared, and utilized. This repository will enhance the availability and accessibility of the software, fostering collaboration and innovation around the UFDS.

Most of the software in the repository will be FOSS (Free and Open-Source Software), as many components are or will be based on FOSS "building materials". However, certain components might remain under non-FOSS licenses due to specific constraints or agreements and might either be included under these licenses or excluded from the repository entirely. It should be noted that ongoing development versions of the above components will also be offered, which will be updated throughout the project as development progresses.

Moreover, the repository will function as a data backend, supporting the user interface(s) associated with the project's app store, connector store, and any other related services. This integration ensures a seamless user experience, allowing users to efficiently discover, download, and utilize the software. An envisioned user interaction with the App and Connector Store is detailed below. The technical aspect of this interaction is described along with the App and Connector store component in the data space's architecture (Section 2.3.4).

User Story: Downloading an app through the UFDS App Store
A user, having previously installed the UFDS core applications (see Section 2.3.4 for details), logs into the App Store UI through their browser. Inside the App Store, they are greeted with a clean, intuitive interface showcasing featured apps organized under categories that can be used to filter the apps. Each application is displayed in its own box, including a short summary of its functionality. A search tool is also available.

Once the user finds the application they are interested in, either by scrolling, filtering, or searching, they click on the app box. This opens a detailed *application page* with comprehensive information about the application.

The application page provides an executable script that the user needs to download and run on their machine. This script then performs the following:

- Downloads or "`git-clones`" the application's source code from the corresponding open software repository which acts here as a database back-end.
- Builds the application using the provided technical blueprints (Dockerfiles).
- Prompts the user to connect using their web-browser to a local url address (e.g., http://localhost:8080) to use the application, also suggesting the user bookmark the url for later use.

A similar executable script to uninstall each application will also be provided.

4.2. Status

The open software repository has been created and can be accessed at https://gitlab.com/disco-horizon-europe. Please note that not all repositories may be public; some repositories might only be viewable by the developers and maintainers. If you require access to these restricted repositories, please contact us.

At the time of writing, the open software repository includes the following:

- **Forks (copies) of various Open Software Repositories:** This includes software related to the Eclipse Data Space Connector (EDC) and the Sovity software packages.
- **The Minimum Viable UFDS (alpha version):** Based on the Sovity connector, this deployment is intended for testing and demonstration purposes. It includes documentation covering both deployment and usage. Once deployed, users can search for and consume datasets provided by Ghent and Thessaloniki either through the REST-API or via the UI. Both options are detailed in the documentation.
- **Active Development Components:** Various components of the UFDS are in active development and are based on the EDC and Sovity software packages. Note that these under-development packages, especially those not yet near their final versions, are likely to be in private repositories and thus not accessible to users without the required privileges.

A screenshot of the Open Software Repository is provided in Figure 16 for illustration reasons.

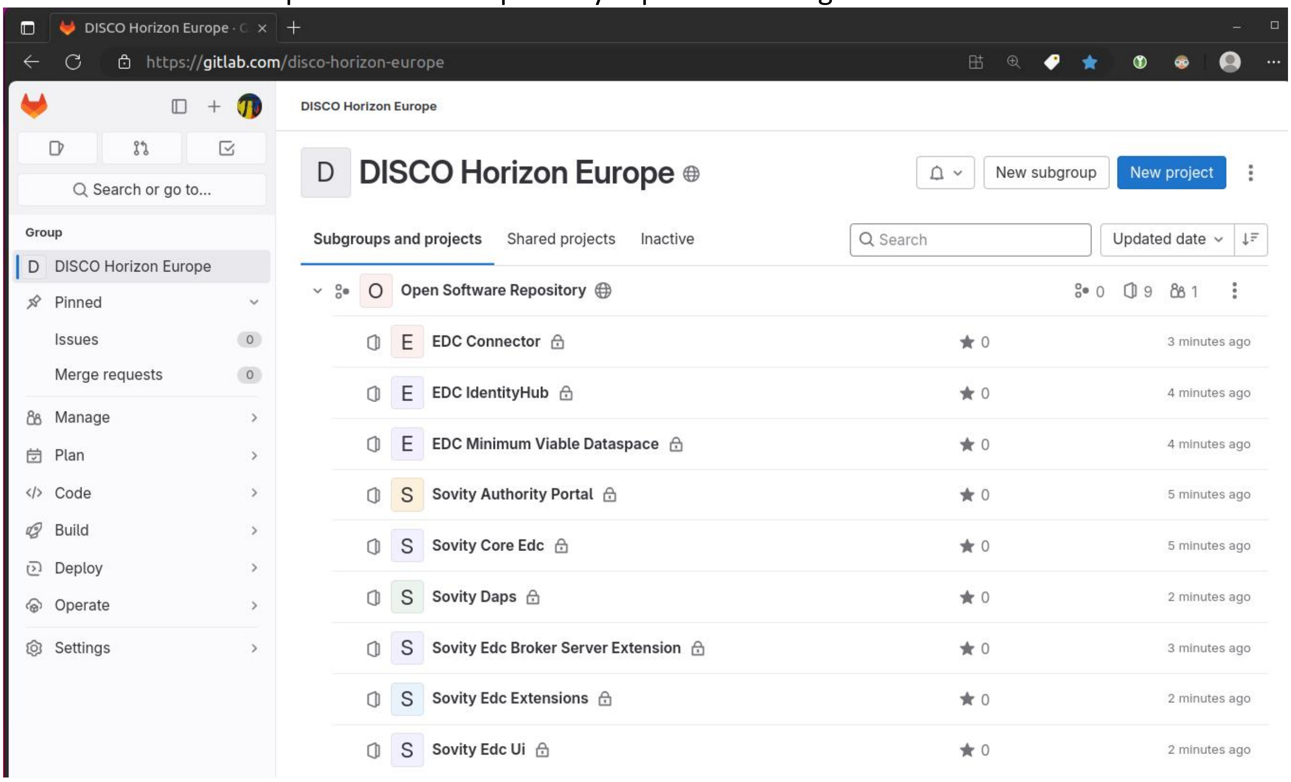

Figure 16: Open Software Repository. Note that not all of the illustrated repositories are publicly available.

As development progresses on the various UFDS components, more repositories will gradually become public. This will include different FOSS applications developed for UFDS. Additionally, the open software repository will serve as the database backend for the App and Connector Store's offerings, which will feature a more intuitive UI for non-technical users.

The next two technical deliverables of WP3 (i.e., D3.2 and D3.4) will provide relevant updates and reports on the above.

5. Conclusion

The Urban Freight Data Space (UFDS) represents a transformative step forward in optimising urban logistics through enhanced data sharing and collaboration. Our project has laid out a comprehensive plan to revolutionize the way the industry's operations are managed, attempting to address the fragmentation in data sharing that currently hampers efficiency and sustainability, by fostering a secure and standardized environment for data exchange, ultimately aiming to break down corporate silos and enable a more integrated approach to urban freight logistics.

Throughout this deliverable, we have detailed the architecture of the UFDS and outlined our strategic development plan, emphasizing the importance of data sovereignty and the need for future-proof solutions. Our approach is designed to address the concerns of private data owners towards sharing their data by ensuring the UFDS platform evolves alongside advancements in data space technology. This will provide the necessary level of data sovereignty and usability to meet the needs of all stakeholders involved.

Our development plan is divided into three key milestones. The first milestone focuses on deploying UFDS Version 1, incorporating city-owned datasets to create a functional platform by October 2024. The second aims to finalize meaningful use cases and enhance the platform with additional datasets and functionalities by Q4 2025, while the final milestone involves documenting the technical advancements needed for a comprehensive urban freight data space that will serve as a cornerstone for future urban logistics data exchange.

As we progress, our iterative process, involving collaboration with WP2 and WP4, will ensure that the UFDS not only meets current needs but also adapts to future demands. This interaction between theoretical models and practical implementation will bridge the gap and ensure that the UFDS becomes an integral part of urban logistics.

In summary, the UFDS is poised to drive efficiency, sustainability, and innovation in urban freight logistics through enhanced data integration and cooperation. By setting a solid foundation now, we pave the way for a future where urban freight operations are optimised and contribute to the overall well-being of our cities.

Appendix: APDS Ontology

In 2018 the British Parking Association, European Parking Association, and the International Parking and Mobility Institute formed the Alliance for Parking Data Standards (APDS), an initiative driven by the pressing need within the parking and curbside management sectors for a unified approach to data exchange. APDS aims to establish a standardized framework for sharing parking-related information globally between parking facilities, mobility service providers, smart city infrastructure, and other stakeholders.

To tackle the complexity of managing parking and curbside resources clear definitions and structured data models are needed. APDS proposes an ontology that serves as a foundational framework to organize and standardize data elements such as parking space availability, pricing structures, session management, and user permissions. This standardization not only enhances operational efficiency but also supports innovation in parking services and urban mobility solutions. Through these common data definitions and structures, APDS enables seamless integration of diverse systems and applications. This interoperability not only reduces integration costs and complexities but also enhances the reliability and scalability of parking-related services worldwide.

Structure

The APDS ontology is structured around a robust data model designed to facilitate data exchange within the parking and curbside management sectors. This structured approach ensures clarity, consistency, and interoperability across diverse applications and systems.
It covers the following range of domains, essential for managing parking and curbside resources effectively:

- **Place:** Central to the ontology, defines physical locations such as on-street and off-street parking spaces, loading zones, and designated areas for specific uses. It incorporates geolocation data to precisely identify each location, supporting detailed spatial management and contextual information crucial for operational decision-making.
- **Rate:** Standardizes pricing structures associated with parking services. It includes definitions for various rate types, such as flat rates, incremental rates, and dynamic pricing models. This domain ensures consistent interpretation and application of pricing information across different systems and jurisdictions, promoting transparency and fairness in pricing.
- **Quote:** Facilitates real-time price quoting and reservation systems; it enables service providers to communicate pricing information effectively and supports dynamic pricing

adjustments based on demand and availability, enhancing user convenience and optimising revenue generation for parking operators.

- **Occupancy:** Tracks the utilization and availability of parking spaces in real-time, providing valuable data for demand forecasting and operational planning. It distinguishes between supply (total available spaces) and demand (current usage), facilitating predictive analytics and efficient space management strategies.
- **Session:** Manages parking events, durations, and transactions, events such as vehicle entry and exit times, duration of stay, and associated payment transactions. Ensures accurate billing and operational insights, supporting seamless user experiences and compliance with parking regulations.
- **Observation:** Captures vehicle presence and behaviour data through sensors and manual checks; utilizes various data collection methods such as license plate recognition (LPR) and RFID tags to support enforcement activities and verify compliance with parking regulations supporting enforcement and compliance efforts.
- **Permissions and Rights:** Defines access permissions and usage rights within parking facilities, ensuring regulatory compliance and efficient management of parking facilities. Includes specifications for rate tables, eligibility criteria, validity periods, and credential management, supporting diverse user scenarios and operational requirements.
- **Equipment and Services:** Encompasses additional functionalities and services within parking facilities, enhancing overall user convenience and operational capabilities. Includes specifications for amenities such as EV charging stations, bicycle racks, and security systems, supporting a wide range of user needs and enhancing overall facility management.

Application on the National Parking Platform (NPP)

One prominent application of the APDS ontology is showcased through the UK Department of Transport's National Parking Platform (NPP).[20] Launched as a pilot in 2021 and scheduled for full rollout by late 2024, the NPP leverages APDS standards to enhance collaboration and data exchange among stakeholders in the parking industry. By integrating APDS-defined domains such as availability, pricing, and session management, the NPP ensures efficient parking services across a wide network of sites. It serves as a centralized hub for managing parking data, supporting functionalities from real-time space availability updates to payment processing and enforcement operations.

[20] https://www.britishparking.co.uk/national-parking-platform

Specifically, APDS role in NPP's platform functionality is two-fold. First, through the NPP acting as a contract manager, enabling parking operators and service providers to become members and participate in data exchange using the APDS standards. This ensures that all participants speak the same data language, facilitating seamless interactions and transactions. Second, parking operators upload their parking information, including locations, tariffs, and operating hours, onto the NPP platform using the APDS ontology. In this way, service providers can then access this data to offer parking services to motorists, such as payment options and session management.