DISCO

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

Deliverable 3.4

DISCO Data Space open software repository Version 2, & Implementation Guide

Panos Protopapas, INLE

30/04/2026

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.

Contents

Abstract

This deliverable presents the final implementation of the Urban Freight Data Space (UFDS), developed within Work Package 3 of the DISCO project. It documents the complete architecture of the platform, encompassing the core infrastructure components, including the data space connector, identity management, federated dataspace capabilities, a publish-subscribe brokering mechanism, logging, and vocabulary management, as well as the suite of data space applications built to deliver operational value to participants. The deployed infrastructure spans more than 20 connectors and 50 datasets, and has been successfully integrated with three other Horizon Europe projects: URBANE, Green.Dat.AI, and GREENLOG. The deliverable further presents the documentation and deployment tooling developed to support future operators and onboarding participants. Beyond the technical record, the deliverable reflects on the conditions necessary for the UFDS to evolve from a project-driven implementation into a durable, sector-wide data sharing infrastructure, addressing both the technical trajectory ahead and the ecosystem dynamics that will determine long-term adoption and impact.

Summary sheet

Deliverable No.	3.4
Project Acronym	DISCO
Full Title	DATA-DRIVEN, INTEGRATED, SYNCROMODAL, 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), Valia Konstantopoulou (INLE), Ioanna Fergadiotou (INLE)
Peer Review	AKKA, IDSA
Quality Assurance Committee Review	FIT
Date	30/04/2024
Status	Final
Dissemination level	Public
Version	0.9
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
JLL	UK	JLL
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	04/03/2026	INLE	Table of Contents	
0.2	10/03/2026	INLE	Architecture	
0.3	15/03/2026	INLE	Implementation	
0.4	17/03/2026	INLE	Documentation and Deployment	
0.5	19/03/2026	INLE	Moving Forward	
0.6	20/03/2026	INLE	Abstract, Introduction, Conclusion	
0.7	23/03/2026	INLE	Peer Review Version	

0.8	30/03/2026	INLE	Reviewers' comments addressed	
0.9	31/03/2026	INLE	Quality Assurance Version	

List of acronyms

UFDS	Urban Freight Data Space
FOSS	Free and Open Source Software
UI	User Interface
PubSub	Publish Subscribe
EDC	Eclipse Dataspace Connector
LSP	Logistics Service Provider
LL	Living Lab
DSP	Dataspace Protocol

2. Introduction

Urban freight logistics is undergoing a fundamental transformation. As cities intensify their push towards net-zero targets, the pressure on LSPs to operate more efficiently, transparently, and collaboratively has never been greater. Data sits at the centre of this transformation, yet its potential remains largely untapped, fragmented across proprietary systems and organisational boundaries. Data spaces have emerged as a principled response to this challenge: a framework for controlled, sovereignty-preserving data sharing that enables collaboration without requiring participants to surrender ownership of their most sensitive assets.

The DISCO project has sought to make this vision concrete for urban freight. Over the course of its duration, Work Package 3 has progressively designed, implemented, and refined the Urban Freight Data Space (UFDS), a purpose-built data sharing infrastructure grounded in European interoperability standards and developed in close alignment with the needs of the urban freight ecosystem. This deliverable represents the culmination of that effort. It is the final technical deliverable of the UFDS development cycle, bringing together the complete architecture, the deployed implementation, and the practical lessons accumulated through real-world operation, cross-project integration, and iterative refinement.

The deliverable covers several interconnected dimensions. Section 3 presents the final UFDS architecture in its entirety, describing both the core infrastructure components (including the data space connector, identity management, federated dataspace capabilities, logging, and vocabulary management) and the suite of data space applications developed to deliver immediate value to participants. Section 4 documents the concrete implementation of this architecture, covering the deployed infrastructure across more than 20 connectors and 50 datasets, and the integrations realised with other Horizon Europe projects, namely URBANE, Green.Dat.AI, and GREENLOG. Section 5 addresses documentation and deployment, presenting the tooling and guides developed to lower the barrier for future operators and participants. Section 6 situates the work within the broader trajectory of data space development, reflecting on the technical and ecosystem dimensions that will determine whether project-driven implementations like UFDS can evolve into durable, sector-wide infrastructure. Finally, Section 7 concludes the deliverable.

Reading this deliverable as a purely technical document would, however, be only half the story. The UFDS was not built in isolation, nor for its own sake. It was conceived as a practical instrument for enabling a new generation of urban freight operations; one in which logistics actors, city authorities, and researchers can exchange data under clear, trusted, and mutually beneficial conditions. The components described in the pages that follow are therefore not merely software artefacts: they are building blocks towards an ecosystem that does not yet fully exist, but whose foundations are now in place. It is in that spirit, at once technically grounded and forward-looking, that this deliverable should be read.

3. Architecture

Section 3 presents the architecture of the data space. It begins with a high-level overview and architectural principles in Section 3.1, followed by a description of the core components in Section 3.2, and concludes with the data space applications in Section 3.3.

3.1. Final Architecture

Figure 1 presents the final high-level architecture of the UFDS, illustrating both the core components and the data space applications, as well as the high-level interactions among the core components. Open-source components are clearly indicated with a green FOSS label, while components that provide a user interface are marked with a blue UI label. All components can be retrieved from DISCO's Open Software Repository[1] (FOSS components) or the Closed Software Repository[2] (non-FOSS components). Of particular interest is the Federated Dataspace Component, which enables the seamless integration of connectors from other data spaces and supports the formation of a Data Space Federation (see Section 3.2.5).

Although the core components and applications of the data space are described in detail in Sections 3.2 and 3.3 respectively, the overall architecture is fundamentally centred around the Data Space Connector, which acts as the main interface for data exchange between participants. In our implementation, the Sovity Connector was chosen (Section 3.2.1), while the user interface is derived from the Sovity UI (Section 3.2.2) and has been adapted through targeted modifications to better align with the needs of the data space. Identity and access management (Section 3.2.3) are handled through Keycloak, based on OpenID Connect (OIDC) and OAuth 2.0, ensuring secure authentication and authorization across participating entities. In addition, all connector-to-connector interactions are recorded through the Logging House (Section 3.2.6) which consists of two sub-components, the Client deployed on the connector side, which forwards the relevant information to the central Logging House Server operated by the data space authority.

Beyond these interaction mechanisms, the architecture also incorporates supporting components that facilitate discovery, interoperability, and operational management. Connectors are automatically informed about the presence of new participants through the Broker PubSub mechanism (Section 3.2.4), enabling dynamic awareness within the ecosystem. When interoperability across multiple dataspaces is required, the Federated Dataspace Component (Section 3.2.5) manages interactions with external identity providers, allowing connectors operating

---

[1] https://gitlab.com/disco-horizon-europe/open-software-repository
[2] https://gitlab.com/disco-horizon-europe/closed-software-repository

under different identity management systems to participate in a federated architecture. Deployment and operational management of the data space, discussed in Section 5, are supported through semi-automated onboarding scripts for Docker-based installations, while Kubernetes-based deployments are handled in a more automated manner through the Data Space Manager and its App and Connector Store (see Deliverable 3.3). Finally, the Vocabulary Hub (Section 3.2.7) provides guidance on recommended data structures and semantic models within the data space, which are used by the various Data Space Applications (Section 3.3) to ensure consistent data interpretation and interoperability.

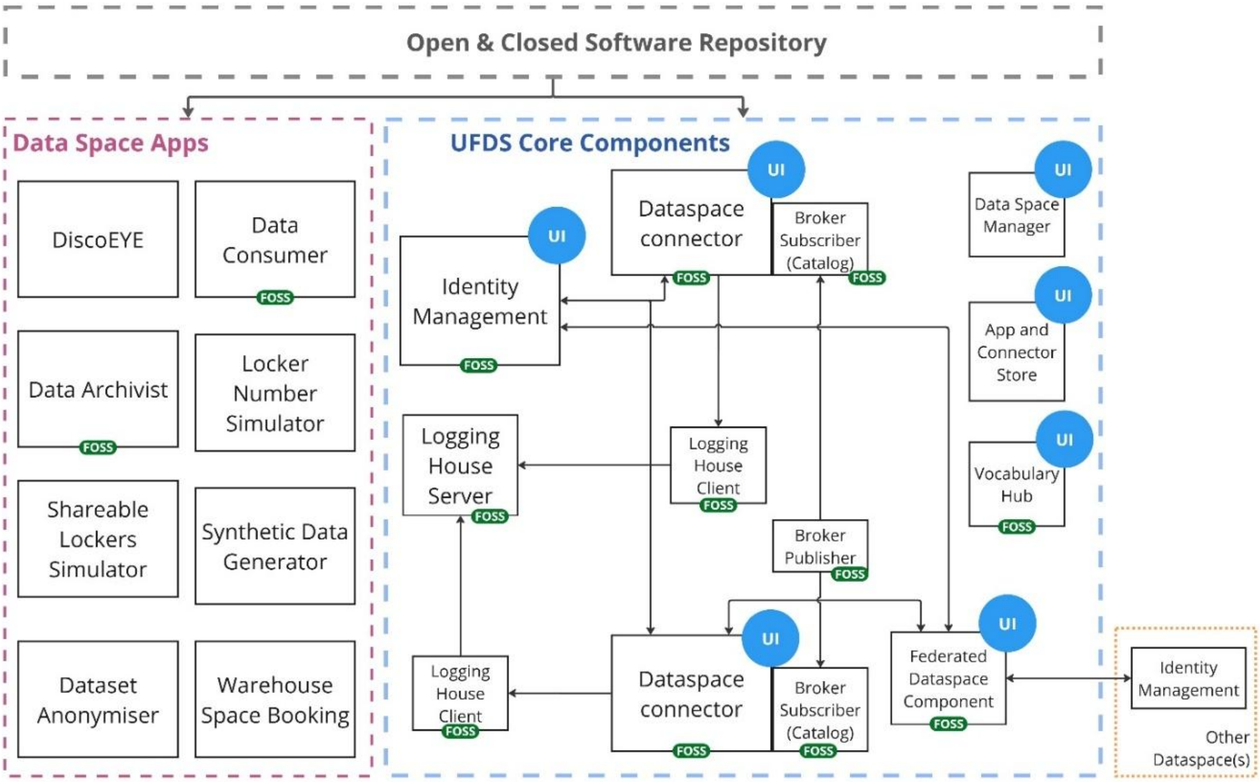

Figure 1: Final UFDS architecture (high-level)

3.2. Data Space Core Components

3.2.1. Data Space Connector

As outlined in Deliverable 3.1, the DISCO project adopted the Sovity Connector for the first version of the data space. During the design of Version 2, the team evaluated whether to continue with Sovity or transition to the Eclipse Data Space Connector (EDC). Since several components had already been developed around Sovity, continuing with it ensured stability and allowed the project to build on the existing integration work.

An additional advantage is that the Sovity Connector is a fork of EDC and therefore shares much of its architecture and functionality, providing a clear pathway for future evolution of the platform. Following Sovity's licensing change at the end of 2024, from the permissive Apache 2.0 license to the more restrictive Elastic License, the DISCO implementation relies on one of the latest Apache 2.0 releases (version 10.4.4). At the same time, future iterations are expected to gradually move toward EDC-based components (see Section 6).

Despite these licensing developments, the Sovity Connector remains a mature and capable solution, well suited for secure data exchange. Its user interface is particularly efficient and easily customizable, which allowed the DISCO team to implement targeted modifications to better support the requirements of the UFDS architecture. Further details can be found in Deliverables 3.1 and 3.2 and on the Sovity Connector's GitHub repository.[3]

3.2.2. Data Space Connector UI

The Sovity Connector UI, described in detail in deliverables D3.1 and D3.2, is used to provide a user-friendly interface for managing connector operations and data sharing activities. Within the DISCO implementation, version 4.1.6 of the UI is employed.

To better integrate the interface within the UFDS ecosystem, several targeted modifications were introduced. These include visual adaptations to incorporate the DISCO logo and theme, as well as functional extensions enabling communication with the Asset Consumer application (Section 3.3.2). Figure 2 illustrates these modifications.

[3] https://github.com/sovity/edc-ce

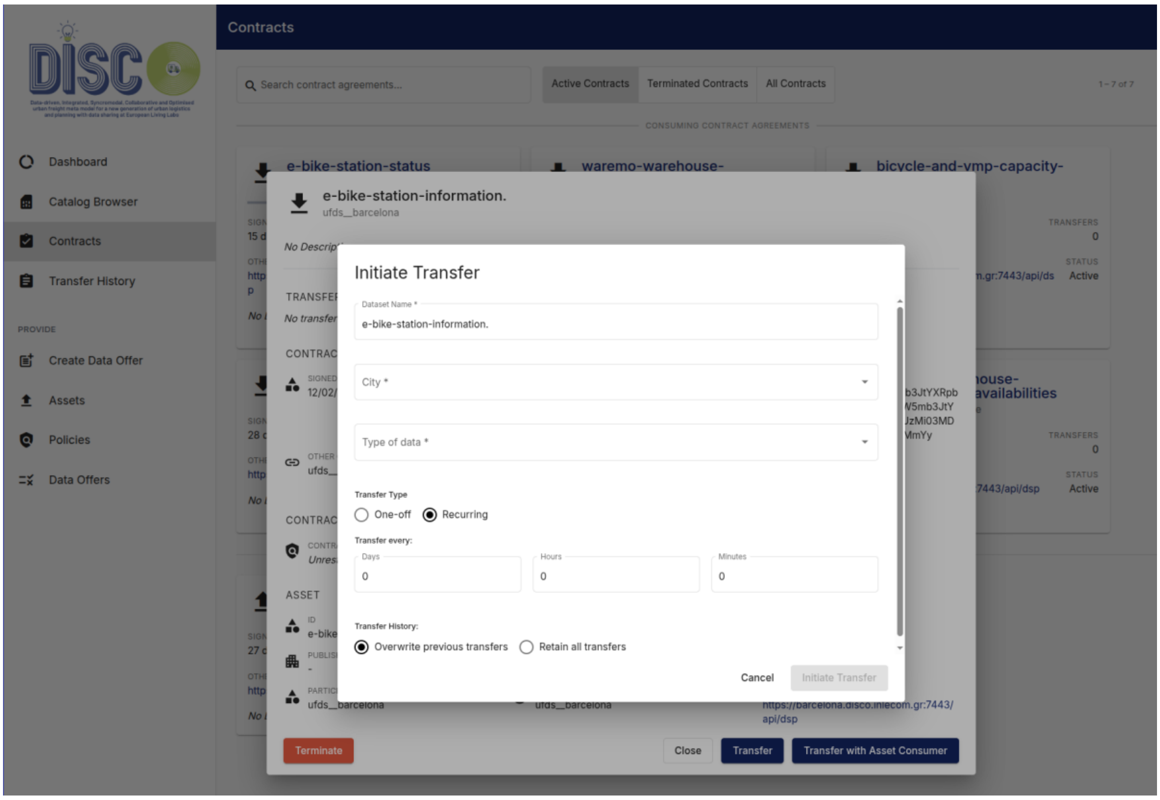

Figure 2: Sovity Connector UI modifications (DISCO theme and communication window with Asset Consumer)

3.2.3. Identity Management

Identity management within the UFDS architecture is handled through Keycloak, which is also the identity management solution used by the Sovity Connector ecosystem. While not a Sovity-developed component itself, Sovity relies on Keycloak for authentication and authorization and provides small modifications to Keycloak's codebase to facilitate its integration with the connector. Both the Keycloak component and the overall approach to identity management in the data space have been previously described in Deliverables 3.1 and 3.2.

In the final architecture, Keycloak fulfils multiple roles. First, it is used to authenticate connectors participating in the data space through mutual TLS (mTLS) with X.509 certificates, managed within a dedicated realm specifically configured for connector identities. In addition, Keycloak is used to protect the public user interface of the Vocabulary Hub. Although this interface is intended to be accessible to all data space participants, access is restricted to authenticated users through a separate realm configured for this purpose. Furthermore, all connector user interfaces that are publicly reachable and deployed within INLECOM's cloud infrastructure are also secured through the same Keycloak deployment, with each connector UI protected via its own dedicated realm. To standardize the security of these web-based interfaces through Keycloak, a lightweight proxy component is utilized as a centralized authentication gateway. This functionality is provided by

OAuth2 Proxy (version 7.6), which intercepts traffic between the user and the protected service. This architectural pattern ensures that both third-party components, such as the Sovity Connector UI (Section 3.2.2) and Semantic Treehouse (Section 3.2.7), and internal services benefit from a uniform authentication layer, ensuring that only authorized users authenticated through Keycloak can access the respective interfaces.

In more comprehensive data space architectures, identity and trust services such as DAPS are typically complemented by a Participant Information Service (ParIS), which acts as a registry enabling participants to discover and retrieve structured metadata about one another, including endpoints, capabilities, and policy-related information. This additional layer supports more dynamic interaction patterns and strengthens governance by extending trust beyond authentication into discoverability and participant-level information management. However, within the scope of the current implementation, the ParIS component has not been materialised. This reflects a deliberate architectural choice aligned with the current maturity level of the data space, where the number of participants remains limited and interactions are relatively well-defined. Under these conditions, the existing identity management approach based on Keycloak and DAPS is sufficient to ensure secure and reliable communication, without introducing the additional complexity associated with maintaining a dedicated participant registry. As the ecosystem evolves and participation scales, the integration of a ParIS component is expected to become a natural extension of the architecture, enhancing onboarding processes, enabling more flexible discovery mechanisms, and supporting the long-term scalability and robustness of the data space.

3.2.4. Data Space Broker and Catalog

In Deliverables D3.1 and D3.2, the architecture initially foresaw the use of the Sovity Data Space Broker as the mechanism for connector discovery and catalog access. However, after further analysis and experimentation with the component, it was concluded that this approach would not fully satisfy the requirements of the UFDS architecture. The Sovity Broker essentially operates as an additional connector within the data space, accessible by all participants, and maintaining a catalog listing datasets that are publicly available to all participants. While functional, this model can only represent datasets accessible by all participants.

For this reason, the broker capability in UFDS has been implemented through a lightweight PubSub mechanism based on two internally developed components: Broker Publisher and Broker Subscriber. The Broker Publisher is operated by the data space authority and maintains the list of active participants in the data space, including each connector's identifier and the public endpoint used to communicate via the Dataspace Protocol (DSP). When a new participant joins the data space, its connector ID and DSP endpoint are added to the Broker Publisher; similarly, when a participant leaves the ecosystem, the corresponding information is removed.

Each deployed connector includes a Broker Subscriber component running alongside it. During its initialisation, the Broker Subscriber initiates a discovery phase by requesting the current list of connectors from the Broker Publisher. This list is stored locally at the connector side and is also passed to the connector user interface through the EDC_UI_CATALOG_URLS environment variable. As a result, the connector UI automatically queries all listed connector endpoints to retrieve information about available datasets. The Broker Subscriber subsequently requests an updated connector list from the Broker Publisher by polling every minute. If changes are detected, the local list is updated and the EDC_UI_CATALOG_URLS variable is refreshed accordingly.

Through this mechanism, all connectors remain continuously informed about the participants present in the data space and can dynamically query their catalogues for available datasets. At the same time, the approach enables connectors to access datasets that may be restricted to specific participants, provided the requesting connector is authorized to do so. Compared to the original broker approach, this implementation offers two key advantages: first, the catalog functionality is effectively integrated within each connector rather than relying on a separate centralized component; and second, the catalog view becomes connector-specific, enabling each participant to see all datasets accessible to them, including those that are not publicly available to the entire data space.

Figure 3 illustrates in a high-level technical flowchart how the Broker component updates the catalogues of the different connectors while Figure 4 illustrates the human-readable view of the Broker Publisher's OpenAPI (Swagger) documentation. The complete openapi.json specification can be retrieved from the publicly available source code in the Open Software Repository. It should be noted that the Broker Subscriber component is fully automated and does not expose an API, as it does not require input from other components or users.

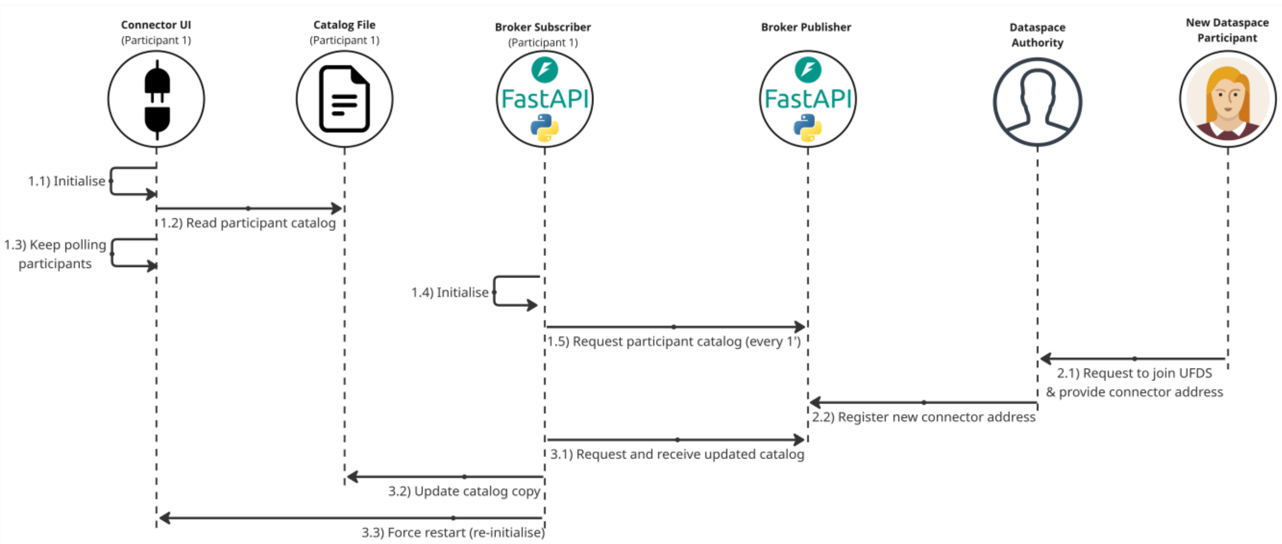

Figure 3: Broker high-level technical flowchart

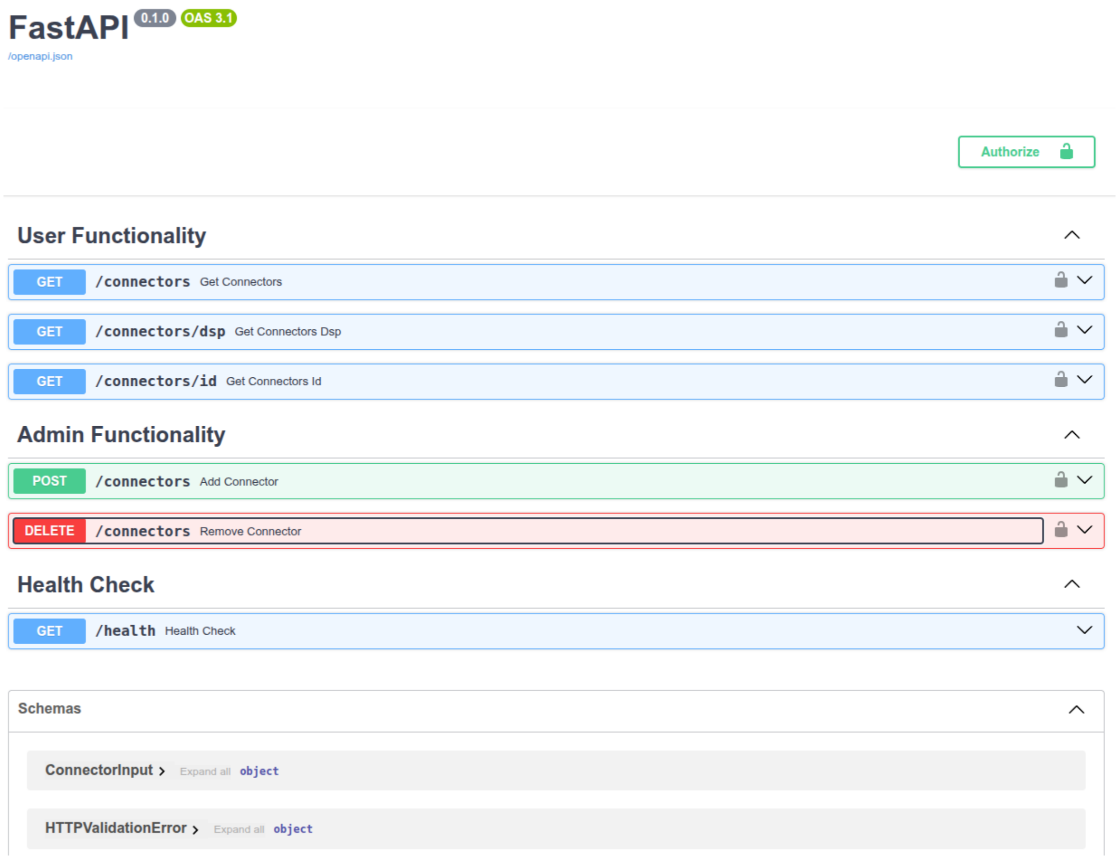

Figure 4: Broker Publisher's OpenAPI

3.2.5. Federated Dataspace Component

The Federated Dataspace Component enables Sovity Connector instances, as well as compatible EDC deployments, to operate across multiple identity management systems and participate in a federated dataspace environment. In the standard configuration, connectors typically trust a single root of trust responsible for validating the full certificate chain used during connector-to-connector communication. This trust is established through standard asymmetric cryptography, where the identity provider (IdP) signs identity tokens using a private key, and participants verify these signatures using public keys typically retrieved from a JSON Web Key Set (JWKS) endpoint. The Federated Dataspace Component extends this model by allowing connectors to trust multiple certificate issuers (i.e., multiple identity management authorities). Crucially, this extension follows the exact same cryptographic principles, relying on asymmetric keys and JWKS-based key distribution, but scales the mechanism to support multiple, independent identity management authorities. In practice, this means that connectors can authenticate and interact with participants belonging to different dataspaces, each operating its own identity provider.

The specific technical implementation of how these asymmetric trust relationships are managed across federated boundaries is described in detail in the accompanying peer-reviewed publication.[4]

Conceptually, this mechanism extends the identity management model from a simple user-based structure (e.g., user 1, user 2) to a more expressive user-and-group-based structure (e.g., user 1 belonging to dataspace A, user 1 belonging to dataspace B). This additional layer of contextual identity allows connectors to establish trust relationships beyond a single dataspace authority while still preserving controlled authentication and authorization mechanisms.

The approach has also been experimentally validated through the federation of two independent dataspaces, specifically between the UFDS ecosystem and the dataspace developed within the Green.Dat.AI project (see Section 4.2.2), with the resulting architecture documented in a published white paper.[5] Within the UFDS deployment, the Federated Dataspace Component is currently used to support interoperability with two additional dataspaces: INLECOM's internal dataspace (Datacore) and the Sovity-based dataspace developed in the GreenLog project (see Sections 4.1 and 4.2.3).

A high-level technical flowchart illustrating how two connectors belonging to different dataspaces can interact using the Federated Dataspace Component is presented in Figure 5. Finally, a basic yet user-friendly interface has been developed. As this component is intended for administrative use only, the interface is intentionally kept simple while remaining sufficiently informative. It provides an overview of the trusted data space keys and allows administrators to add new keys or remove existing ones. The user interface is illustrated in Figure 6.

Last but not least, the component is also listed on the Data Spaces Support Centre's Toolbox, a collection of software components, standards, and guidelines that enable organizations to build, deploy, and operate interoperable and sovereign data spaces.[6]

---

[4] Protopapas, Panos. "Towards interconnected dataspaces: implementing decentralised identity management via DAPS." *The Third International Workshop on Semantics in Dataspaces, co-located with the Extended Semantic Web Conference*. 2025.

[5] Protopapas, P., et al. Connecting Data Spaces, a Practical Approach Using the Sovity Connector. 1.1, Zenodo, 23 Dec. 2025, https://doi.org/10.5281/zenodo.18030960.

[6] https://toolbox.dssc.eu/?pane=tools&service=business-and-org-tools&business=intermediaries-and-operators&tool=sovity-connector-plugin-data-space-federation&tool-details=sovity-connector-plugin-data-space-federation

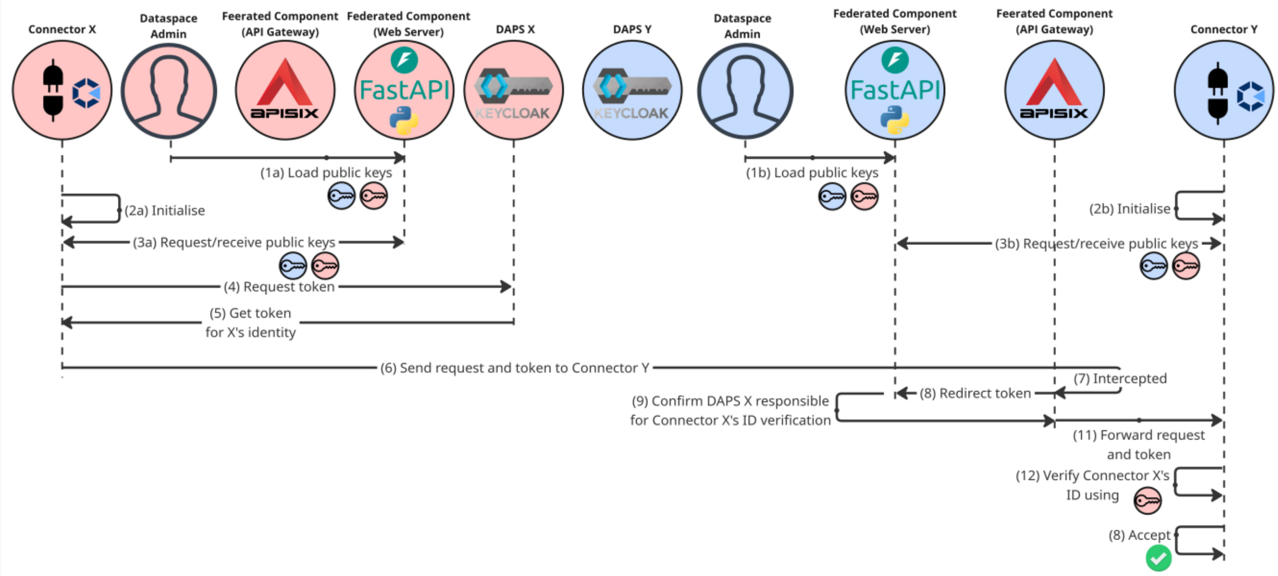

Figure 5: Federated Dataspace Component - high-level technical flowchart

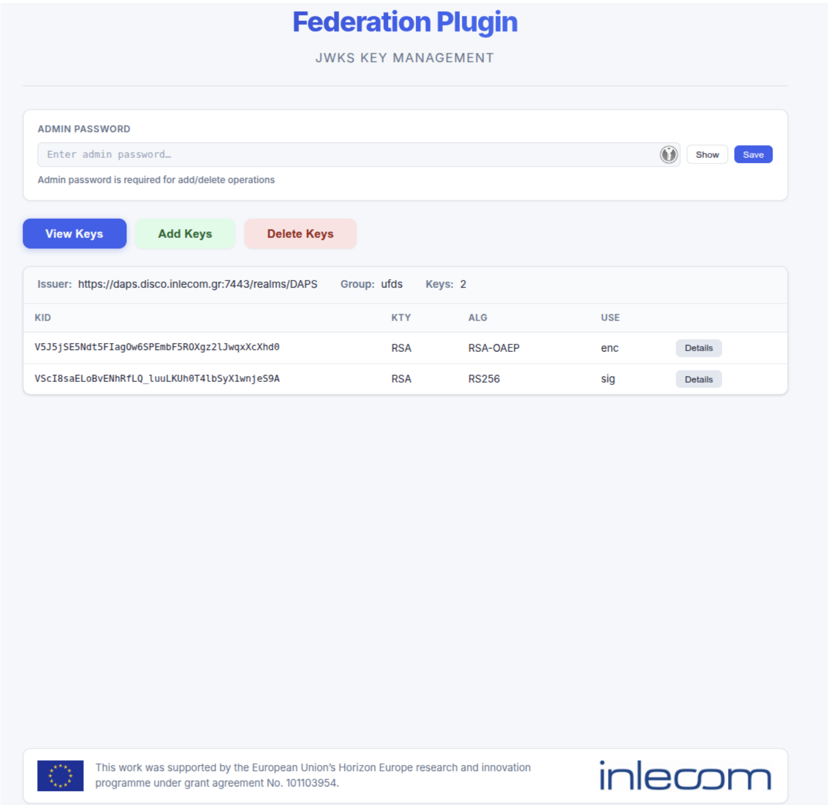

Figure 6: Federated Dataspace Component - User Interface

3.2.6. Logging House

The Logging House component is responsible for recording all connector-to-connector interactions within the data space. It consists of two parts: the Logging House Client, which is deployed alongside each connector and is responsible for registering interaction events, and the Logging House Server, which is centrally deployed and aggregates the logs received from all clients. The collected logs can be queried by connectors, but access is restricted to records that involve the requesting connector; however, an admin role with the ability to query all collected logs can be provisioned. The Logging House component and its functionality have been fully described in Deliverable D3.2, where additional technical details are provided. Figure 7 below illustrates the OpenAPI specification of the Logging House Server. As it is largely identical, the corresponding specification for the Logging House Client is not included here.

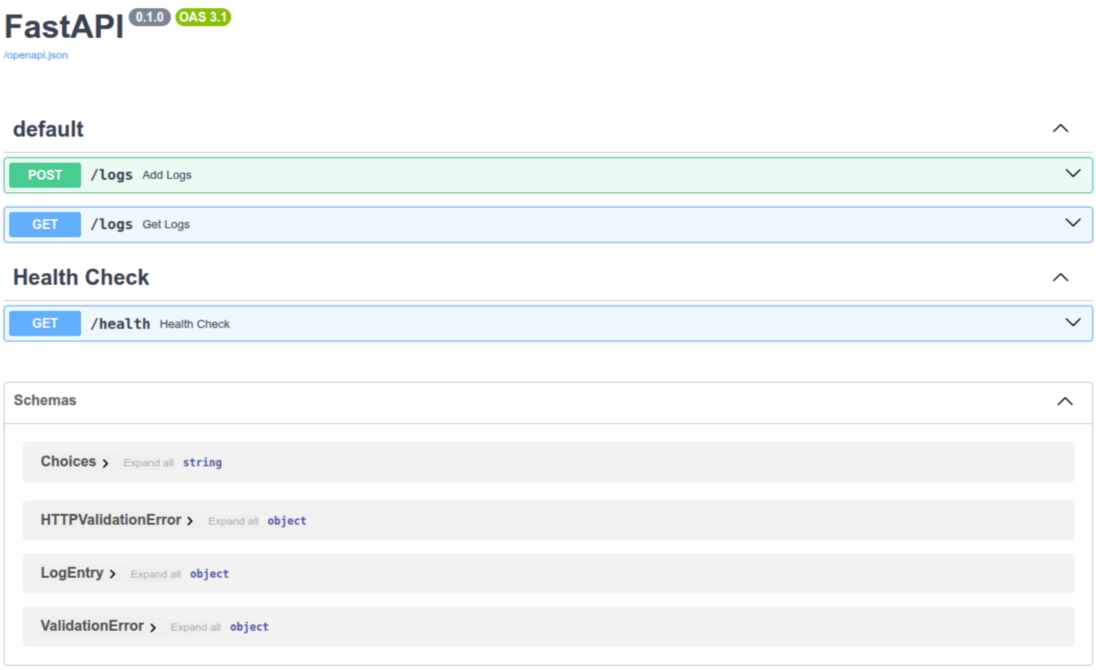

Figure 7: OpenAPI specification of Logging House Server

3.2.7. Vocabulary Hub

The Vocabulary Hub serves as the central repository for documentation on the main data structures recommended for use within the data space. During the development reported in Deliverable D3.2, the implementation was initially based on MkDocs. However, a re-evaluation of TNO's Semantic Treehouse[7] (first considered in D3.1 but previously set aside due to deployment limitations) showed

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

that version 3.6 of the software fully met the project's requirements. Semantic Treehouse is a purpose-built vocabulary management and semantic modelling platform, designed to support the collaborative development, versioning, and governance of data models and ontologies in data spaces. By forking Semantic Treehouse v3.6 and introducing configuration changes, the team enabled two user roles: admin users, representing the data space authority with full read/write privileges, and participant users, who can browse and consult the vocabularies in read-only mode.

Semantic Treehouse brings several advantages to the dataspace. Its structured approach allows participants to browse, validate, and generate machine-readable artefacts, including JSON Schema, XML Schema, and API specifications, directly from curated vocabularies. This ensures that all participants work from a shared semantic foundation, reducing the risk of inconsistencies and promoting interoperability across applications. The Vocabulary Hub leverages this capability to provide guidance on key data structures such as GeoJSON, the EPA-APDS ontology, and Datex II. While these structures are recommended rather than enforced, they act as a "soft standard", particularly for applications handling geolocated or domain-specific data, providing both flexibility for new participants and consistency for application development.

In D3.1, the project envisioned providing a diff-like feature, enabling participants to compare their own data structures against the recommended vocabularies to identify misalignments. While Semantic Treehouse does not natively provide such a JSON-diff functionality, it does allow validation of instances against the registered schemas, offering a partial but meaningful way to detect structural inconsistencies. This approach ensures that participants can still assess alignment with the official vocabularies.

Looking ahead, future updates, potentially extending beyond the project's lifetime, will continue to deliver applications compatible with these suggested data structures. Moreover, the platform's semantic modelling foundation opens the door to more advanced support for participants wishing to restructure their datasets to align with the recommended standards, for instance by leveraging tools such as IMEC's Smart Data Platform (see D2.4: DISCO-X Enabling Tools). In adopting Semantic Treehouse, the data space benefits from a scalable, community-driven vocabulary management solution that enhances clarity, interoperability, and long-term maintainability, while still providing avenues for alignment verification through schema-based validation.

Figure 8 illustrates the Vocabulary Hub where users can access information on the DATEX II, APDS, and GeoJSON data structures.

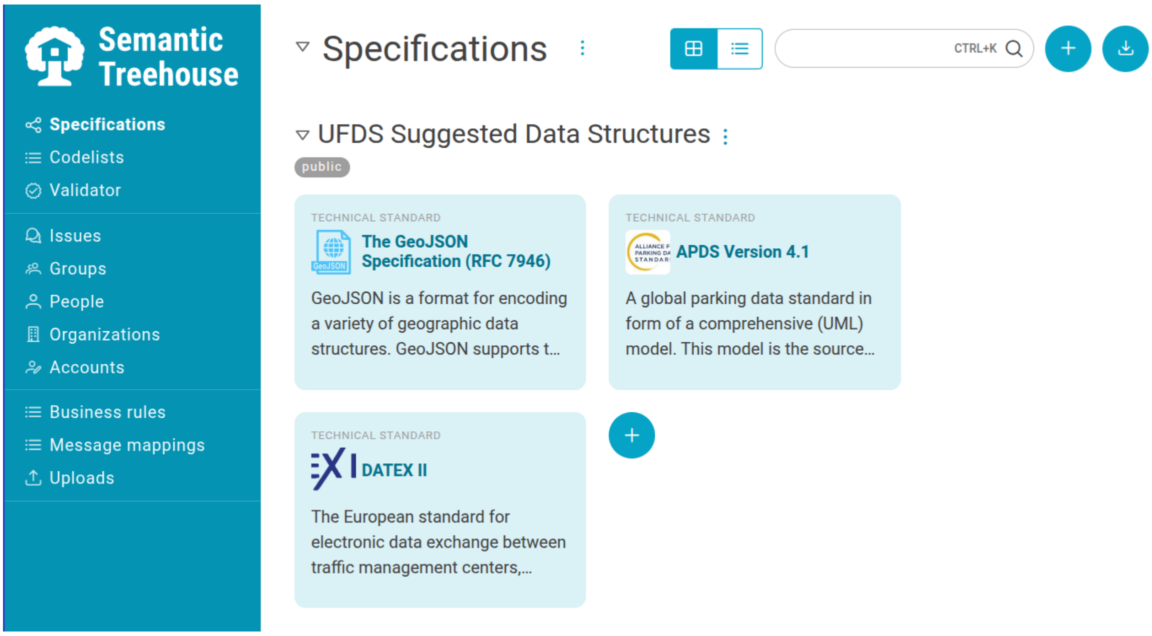

Figure 8: Vocabulary Hub (admin interface)

3.3. Data Space Applications

3.3.1. DiscoEYE

DiscoEYE was finalised before Month 18 of the project and has been fully documented in Deliverable D3.2. The system is composed of three separate applications: (1) a Control application that communicates with the data space connector to consume datasets at predefined scheduled intervals, (2) an Adapter application that performs on-the-fly transformation of heterogeneous non-GeoJSON inputs into the GeoJSON format, and (3) a Visualiser application that displays the resulting datasets on a map interface. The first two applications are open source, while the Visualiser application remains closed source. Furthermore, the Control and Adapter applications are now considered legacy components, as their functionality is more effectively covered by the Asset Consumer (Section 3.3.2) and Data Archivist (Section 3.3.3) applications, respectively, which provide a more robust and integrated approach within the current system architecture.

3.3.2. Asset Consumer

The Asset Consumer component facilitates the one-off or recurring consumption of datasets from the data space. It allows participants to schedule dataset retrieval at specific intervals and integrates with the Data Archivist (Section 3.3.3) to store the retrieved data locally on the consumer's side. The automation of these tasks is managed via cronjobs, which execute a standardized Python script designed to fetch data assets from the connector and transfer them to the Data Archivist.

Through this interface, users can configure the parameters of the fetch, such as the specific data asset ID and destination, which are then passed to the automated command. The schedule of these commands is managed by the system's cron service, ensuring that the standardized retrieval script runs at the user-defined intervals without manual intervention. Once a data asset is retrieved, its contents are stored locally and can be immediately accessed by other components or services within the participant's network. The corresponding Connector UI extension is illustrated in Figure 2 above, while Figure 9 below presents the OpenAPI specification of the Asset Consumer.

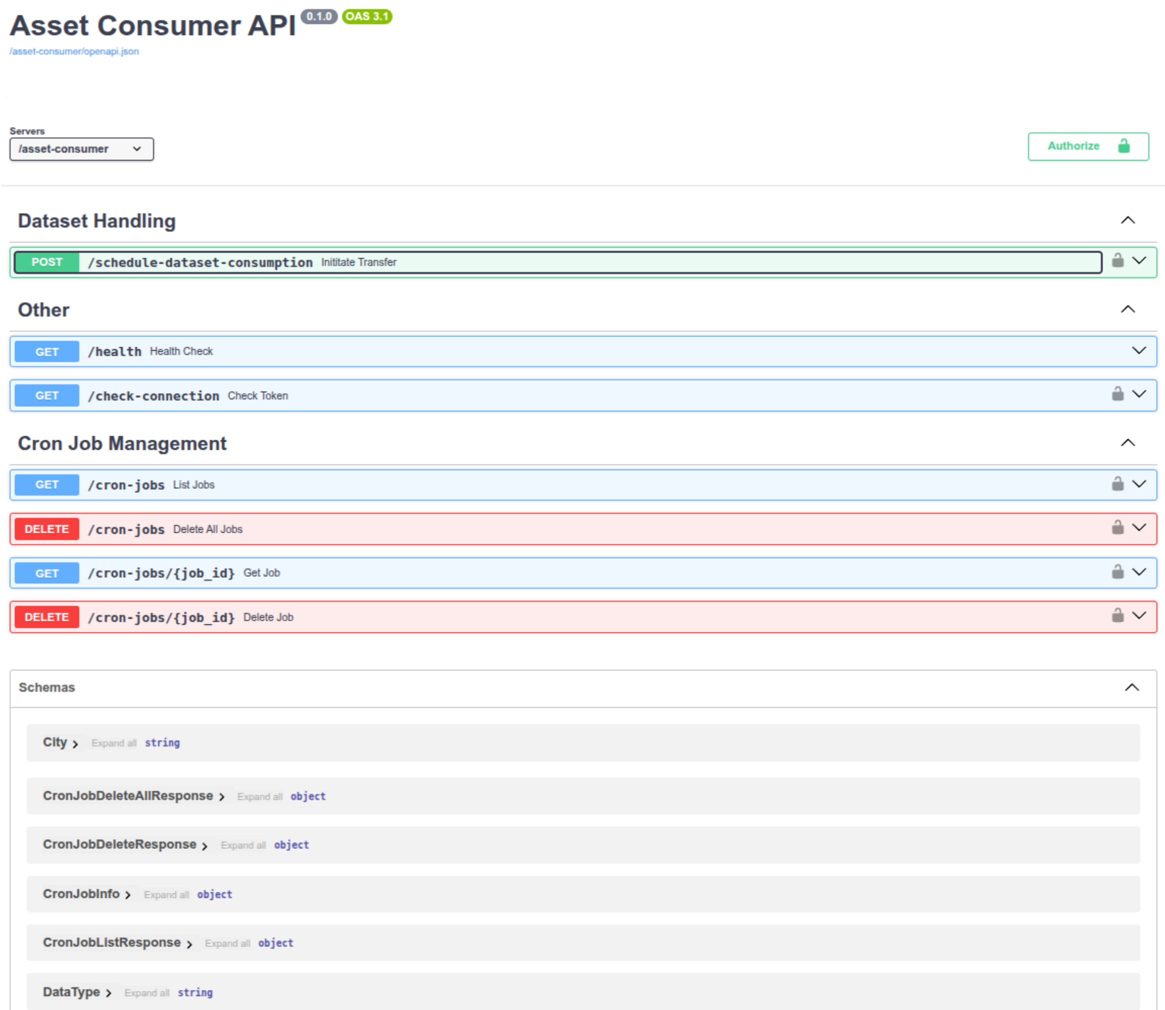

Figure 9: Asset Consumer - OpenAPI specification

3.3.3. Data Archivist

The Data Archivist component acts as the primary data storage endpoint for datasets consumed from the data space. It is designed to be contacted by other connectors during dataset transfers, for the data to be pushed there, and therefore needs to be publicly accessible. Access to the component is secured through a Bearer Token authentication mechanism (API key) to ensure that only authorized participants can store or retrieve data.

Within the architecture, the Data Archivist serves as the data sink configured by the Asset Consumer when datasets are retrieved from the data space. It supports multiple dataset types, including JSON, GeoJSON, and binary data, and maintains a historical record of all consumptions based on the time at which the data was retrieved. The component is backed by MongoDB for persistent storage. Its implementation has been designed to be extensible, allowing additional data types to be supported in the future and enabling the potential integration of alternative storage systems that may be better suited for specific types of data.

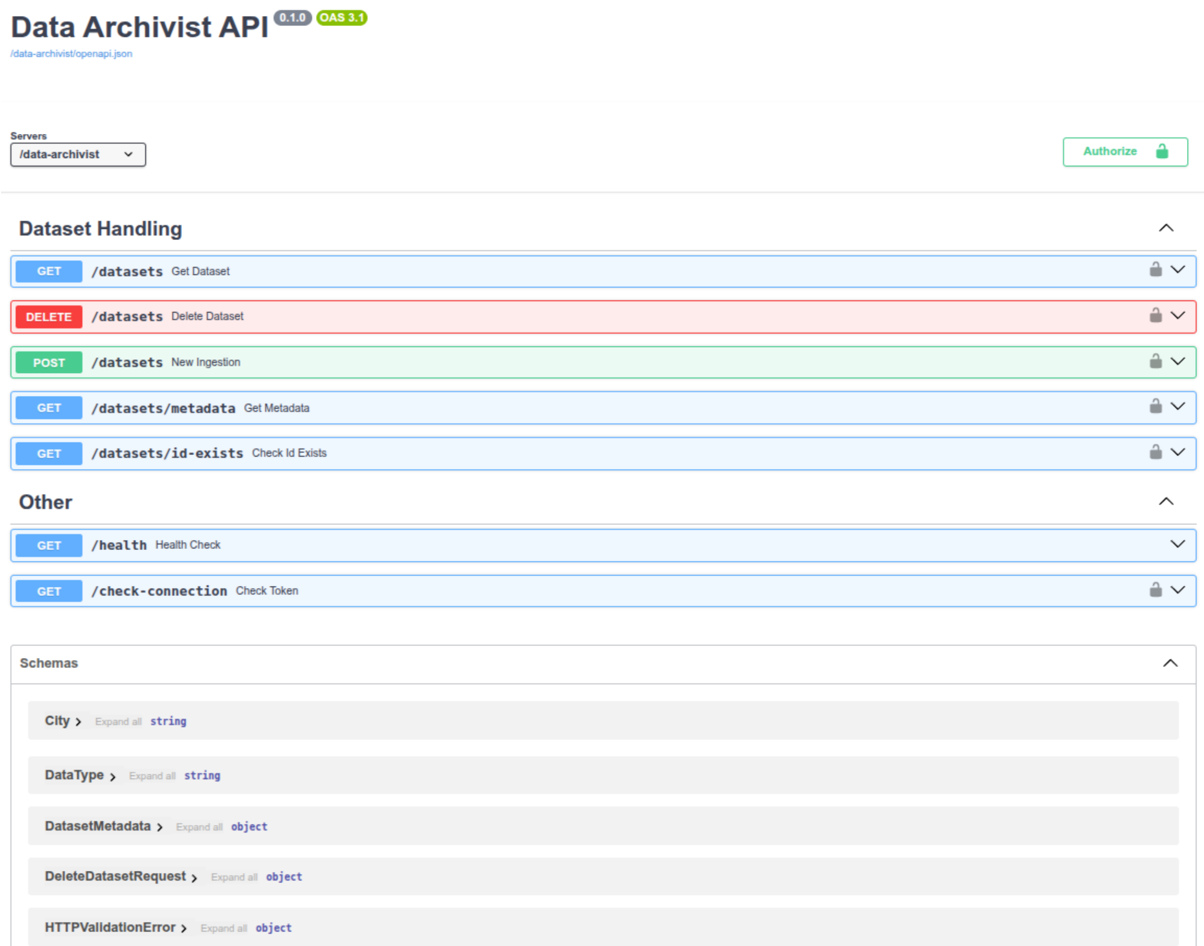

Figure 10: Data Archivist - OpenAPI specification

In addition to basic dataset storage, the Data Archivist provides flexible retrieval mechanisms. Data can be queried not only by dataset identifier but also by data type and time of consumption, enabling users to easily retrieve data consumed at specific intervals.

Finally, another usage of the Data Archivist is to act as a database from which datasets can be provided to the connector for sharing on the dataspace, via a REST API connection. The OpenAPI specification of the component is illustrated in Figure 10 above.

3.3.4. Data Anonymisation

The Data Anonymisation application, developed by IRT SystemX as part of the DISCO Copenhagen LL, provides a structured pipeline to ingest location data, link it to a defined study area, and generate anonymised outputs suitable for downstream use. More information about this LL is available in DISCO work package 4 deliverables, namely D4.1, D4.2, and D4.3.

Users provide input data containing key information such as longitude and latitude. Using the Copenhagen study area and a road network from OpenStreetMap, the pipeline prepares the geographic context and transforms the data into aggregated, anonymised outputs, including average stop counts on configurable hexagon grids or flow counts on road segments. Processing is orchestrated via a workflow engine that ensures each step runs sequentially and data is properly anonymised before finalisation.

The anonymised outputs can then be ingested by other data space components (e.g., stored in the Data Archivist or published via the Data Space Connector) to enable safe downstream consumption and analysis. This application is therefore essential for converting sensitive transport or trajectory data into high-level aggregated forms compatible with the wider DISCO ecosystem.

Although the pipeline's core functionality has been finalised and available in DISCO's Open Software Repository since early 2025, integration with the broader data space ecosystem is ongoing and expected in summer 2026. This also affects the GeoJSON Data Synthesizer and Anonymiser, effectively the UI for this application, developed with the URBANE project and the CITIQORE platform (see Section 3.3.6). A final report with illustrations or video demonstrations will be provided in Deliverable 7.4: Liaison with other initiatives and ICBT.

3.3.5. Warehouse Space Booking (Ware M&O – DISCOESTATE)

The Warehouse Space Booking application is provided via CERTH's Ware M&O platform, which offers a warehouse-as-a-service solution implementing the DISCOESTATE model. It interfaces with UFDS to receive real-time updates of available warehouse slots.

For demonstration purposes, the UFDS connectors in Thessaloniki share warehouse availabilities at the Thessaloniki International Fair (TIF) location to the data space. This availability dataset is consumed via the Meta Model Suite every minute, ensuring that the data copy stored in the Data Archivist instance is always up-to-date. The Ware M&O application then uses this dataset to display available warehouse capacities to users.

When a user books some capacity on a warehouse for a specific timeslot, the booking is confirmed against the original database maintained by Thessaloniki, forcing an update in the data copy stored in the Data Archivist, resulting in the change being immediately reflected in the Ware M&O application.

Although currently a proof-of-concept, this use case demonstrates that as additional UFDS participants share warehouse availabilities through the data space, these can be seamlessly integrated into the platform, enabling users to browse and book warehouse spaces from multiple locations within a single application.

For more details on the Ware M&O platform, see Deliverable D2.4: DISCO-X Enabling Tools, Section 5.1. Information on the Meta Model Suite architecture and the DISCOESTATE model is provided in Deliverable D2.3: DISCO Meta Model Suite Architecture.

The data pipeline of the architecture behind the Ware M&O application is illustrated in Figure 11.

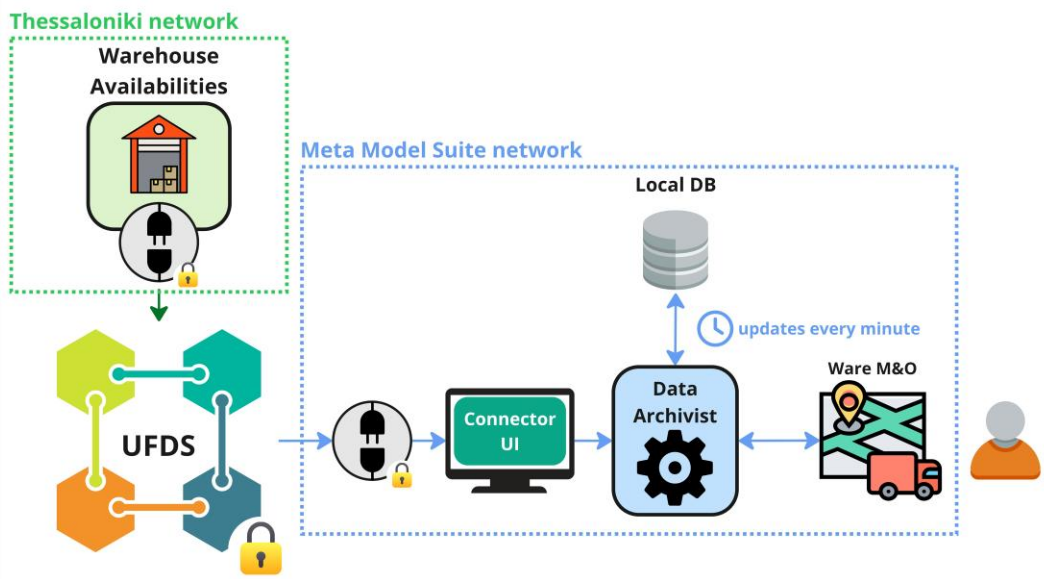

Figure 11: Data pipeline of Warehouse Space Booking application (WARE M&O - DISCOESTATE)

3.3.6. DISCO-URBANE Applications

URBANE Project (Grant Agreement No. 101069782) is an EU-funded initiative focusing on advancing sustainable urban logistics through data-driven and AI-enabled solutions. For a more high-level overview of the integration purpose and results between DISCO and URBANE, the reader is referred to Section 4.2.1.

The three applications presented below, (1) Optimal Locker Number Simulator, (2) Shareable Lockers Simulator, and (3) GeoJSON Data Synthesizer and Anonymiser, are the outcome of this collaboration. All three applications make use of the URBANE User Interface, namely the CITIQORE application, which serves as the primary interaction point for end users. Additional information regarding CITIQORE can be found in URBANE's D3.4: Innovation Transferability Platform documentation.

All applications are designed as data space-enabled tools that allow stakeholders to either test different operational scenarios using LSP datasets or to create GeoJSON datasets, either from scratch or by anonymising existing ones, to be used as needed. The user interface is implemented through the URBANE UI platform enabling users to connect each application to their own Data Archivist and Data Anonymisation instances.

The data pipeline across all three applications follows a largely common approach. To establish connectivity, the user provides the root API endpoint of the Data Archivist (for the first two applications) or the Data Anonymisation App (for the third application), along with a personal access token for authentication. This information is processed exclusively within the user's browser and is neither transmitted to nor stored by the URBANE platform. Through this mechanism, the CITIQORE application communicates directly with the Data Archivist to retrieve datasets, and with the Data Anonymiser to request either the anonymisation of existing datasets or the generation of synthetic ones.

Regarding dataset retrieval, it is important to emphasize that all datasets used by the applications are fetched directly from the user's Data Archivist instance, where they have previously been stored following their consumption via the data space connector. This design ensures that even restricted datasets remain fully under the control of their respective asset consumers. The datasets are used only during the active user session, are not uploaded to the CITIQORE cloud infrastructure, and are not accessible by the URBANE platform. This guarantees the preservation of data sovereignty, which is a fundamental principle of data spaces.

Finally, all applications operate using the GeoJSON format, either as input (in the case of the first two applications) or as output (in the case of the third application).

An overview of the data pipeline, from the publication of a delivery dataset in the data space via a Connector, to its consumption, storage in the Data Archivist, and subsequent utilisation by one of the first two applications, is illustrated in Figure 12. Similarly, the pipeline supporting the anonymisation of an existing dataset within the data space is presented in Figure 13.

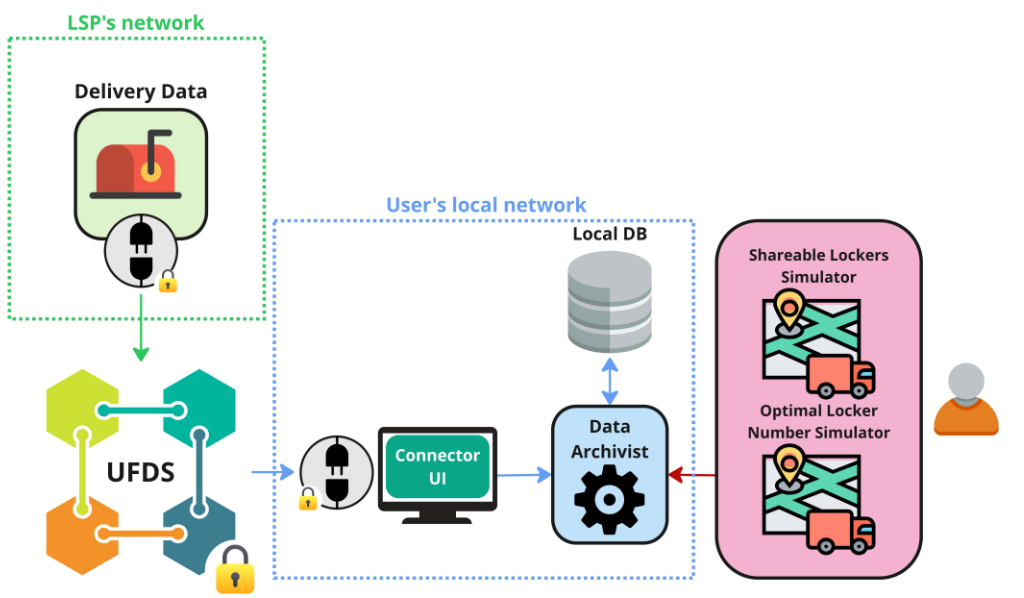

Figure 12: Data pipeline of Optimal Locker Number Simulator and Shareable Lockers Simulator

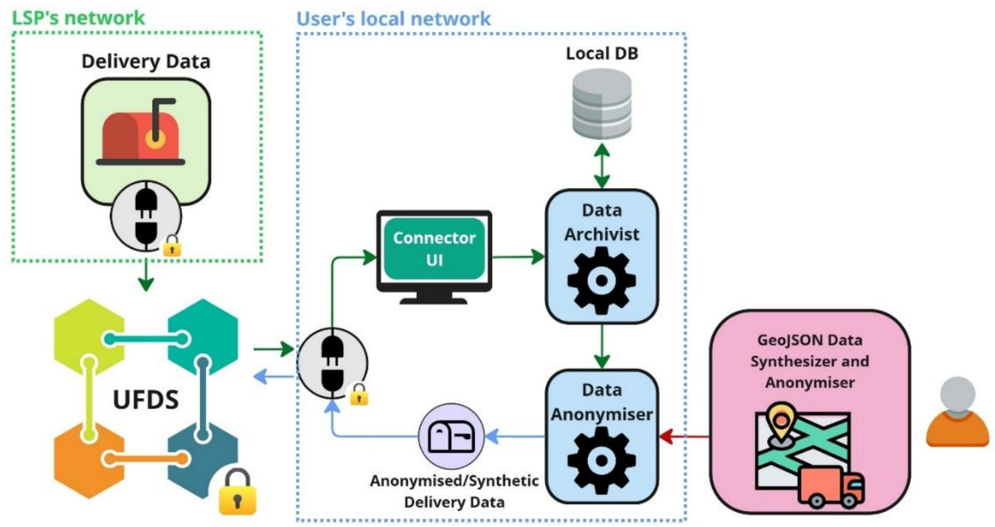

Figure 13: Data pipeline of GeoJSON Data Synthesizer and Anonymiser

Optimal Locker Number Simulator

In this application the user is required to provide the connection details to their Data Archivist instance (URL and access token). Once a connection is established, the user selects the city and a door-to-door LSP delivery dataset to be analysed. Following this, the user defines hypothetical locations for parcel lockers, enabling the simulation of a transition from a purely door-to-door delivery model to a hybrid model where some deliveries are redirected to nearby lockers instead of the customer's address.

Following the execution of the scenario, the simulator calculates and presents several performance indicators, including the number of parcels served through lockers, the distances customers must travel to retrieve their parcels, and the estimated CO_2 emissions associated with the simulated delivery operations.

The application also supports the execution of multiple scenarios, which can be compared side-by-side to evaluate different locker placement strategies and identify more efficient configurations.

A video demonstration of the application is available online.[8]

Shareable Lockers Simulator

In this application the user is required to provide the connection details to their Data Archivist instance (URL and access token). Once a connection is established, the user selects the city, one door-to-door LSP delivery dataset (e.g., Deliveries of LSP 1), and two LSP locker locations datasets (e.g., Locker locations of LSP 1 and Locker locations of LSP 2).

The application then executes a what-if simulation consisting of two sub-scenarios. In the first scenario, LSP 1 performs deliveries using only its own lockers (i.e., Locker locations of LSP 1). In the second scenario, lockers are treated as a shared pool, allowing deliveries from LSP 1 to use any available locker (i.e., both Locker locations of LSP 1 and Locker locations of LSP 2).

The simulation produces statistics related to the number of packages served via lockers, the distances customers travel to collect their packages, and the estimated CO_2 emissions generated by locker deliveries. More importantly, the results of the two scenarios are presented side by side to enable the direct comparison between the effects of operating under an independent and a shared locker infrastructure.

An illustration of the application's user interface is shown in Figure 14.

[8] https://www.youtube.com/watch?v=1synW7s6gf8

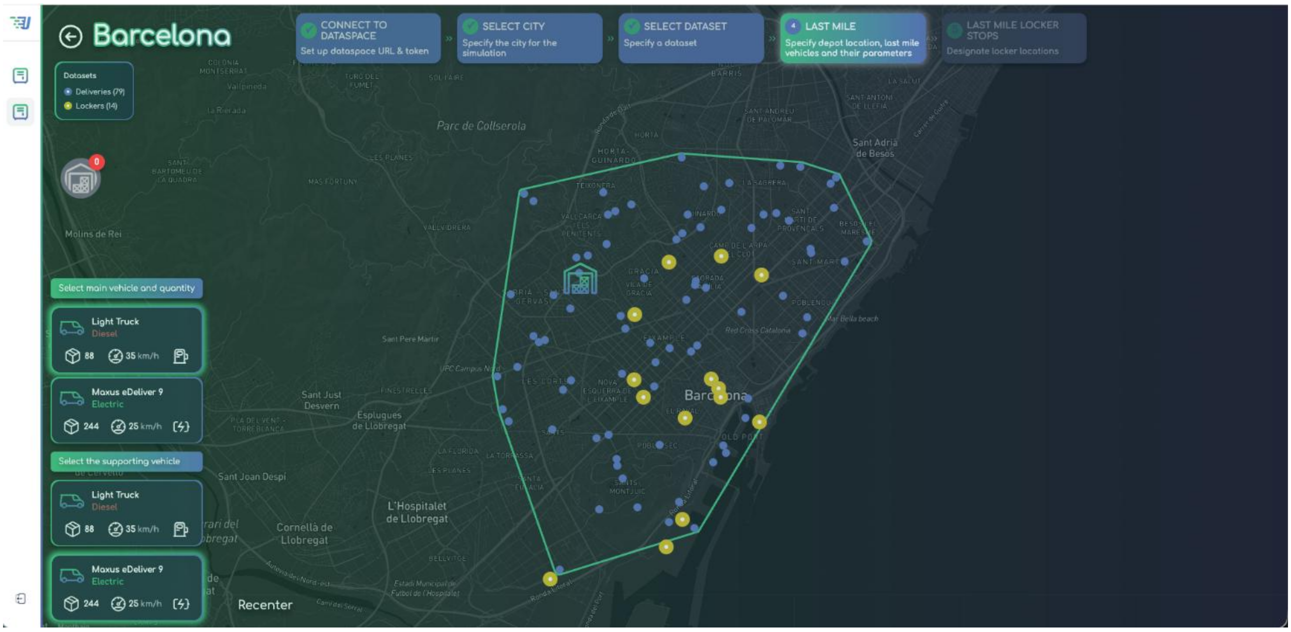

Figure 14: Shareable Lockers Simulator interface

GeoJSON Data Synthesizer and Anonymiser

The GeoJSON Data Synthesizer and Anonymiser allows users to either anonymise existing datasets retrieved via the data space or generate new GeoJSON datasets.

The user first provides the connection details to the Data Anonymiser (API endpoint and access token). After connecting, they select a city and choose between anonymising an existing dataset or creating a new one. In both cases, the desired number of output points is defined. For anonymisation, the user additionally sets the Anonymization Strength, which controls the average displacement of points (with 0 corresponding to no anonymisation). For dataset synthesis, the user selects a bounding box on the map, chooses a generation method (uniform or centroid-based), and configures relevant parameters such as centrality.

Once configured, the request is sent to the Data Anonymisation application. This component interacts with the Data Archivist, retrieving data in the case of anonymisation or storing the generated dataset, and subsequently with the data space connector to publish the resulting dataset as a new data asset, enabling the user to create a corresponding data offer.

As noted in Section 3.3.4, the functionality for this application is fully implemented from the CITIQORE side, and the dataset anonymisation and synthesis features have been developed on the UFDS applications side. The remaining work, completing the integration with the Data Archivist to store newly created datasets and with the Data Space Connector to publish them, is expected to be finalised by late spring 2026. A final update on the application, including relevant illustrations and/or a video demonstration, will be provided in Deliverable 7.4: Liaison with other initiatives and ICBT.

4. Implementation

The implementation of the UFDS has been carried out entirely on Inlecom's cloud infrastructure. Nevertheless, deployment on the connector owners' own infrastructure was both supported and offered as an alternative.

Owing to the comprehensive documentation and deployment material provided (see Section 5), the system is designed to be easily deployable across different environments. . Following the deployment of each connector, initial secure credentials were provisioned through Keycloak for both the connector User Interface and the Vocabulary Hub. To ensure security from the outset, the system utilizes Keycloak's native credential management policies, which require users to perform a mandatory password reset upon their first login. In addition, strong, randomly generated passwords were assigned to all other internal components deployed within each connector's environment. The initial credentials for the connector UI and Vocabulary Hub were securely communicated to each connector owner. In total, the UFDS comprises more than 20 connectors and over 50 datasets, with the overall deployed architecture as well as the full list of connectors and datasets provided in Section 4.1.

Furthermore, Section 4.2 presents the integrations achieved with other projects. These include direct integrations with the UFDS, namely URBANE (Section 4.2.1) and Green.Dat.AI (Section 4.2.2) as well as indirect integration, with UFDS components being used by GREENLOG (Section 4.2.3).

4.1. Deployed architecture

The deployed architecture consists of three distinct data spaces, deployed using the Docker Compose method, forming a federated ecosystem: the UFDS, Inlecom's DataCore data space, and Frontier's EDC-based data space. Together, these deployments demonstrate interoperability across heterogeneous project environments.

An overview of the combined architecture is illustrated in Figure 15. Note that, for clarity and space considerations, certain components, such as the connector UI, Logging House client, Broker Subscriber, and data space applications alongside each connector, are omitted from the figure.

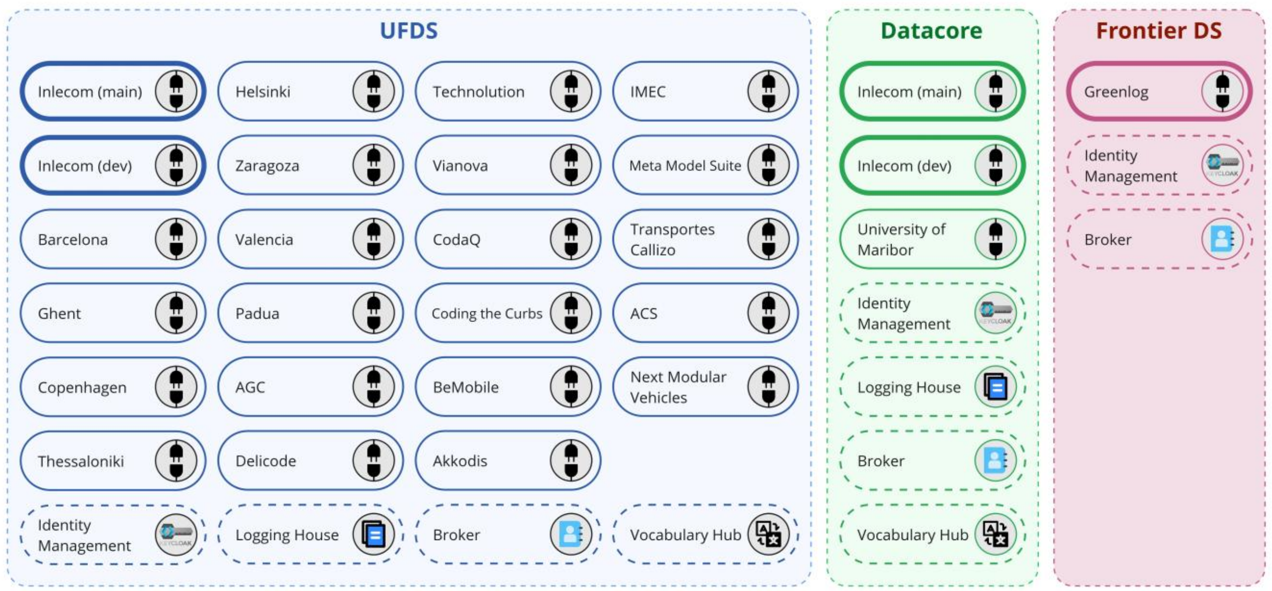

Figure 15: Deployed architecture (simplified)

The UFDS constitutes the largest deployment of the three data spaces, comprising a total of 23 connectors and 57 data assets. Complementing this, DataCore represents Inlecom's most recent data space deployment, designed to unify data spaces across different projects through the use of the Data Space Federation Plugin and become in the future the central pillar for development, testing, and deployment of all data space related components. At the time of writing, DataCore includes 3 connectors and 6 datasets.

In addition, a third data space has been deployed by Frontier based on EDC technology. This deployment is primarily used as a proof-of-concept to demonstrate interoperability between data assets originating from the GREENLOG project, which normally leverages FIWARE-based connectors, and analytics models developed within the URBANE project on the CITIQORE platform, which are currently compatible with EDC-based connectors. As such, this data space remains limited in scope, containing a single connector and 3 data assets. More information on the GREENLOG integration using UFDS components can be found in Section 4.2.3.

Interoperability across these data spaces is achieved through the Data Space Federation component, which enables connectors from different data spaces to communicate with one another and operate on a federated data space level. Importantly, federation is established at the connector level rather than the data space level. This means that individual connectors can selectively participate in one or more federations, and not all connectors are necessarily part of the same federation. As a result, multiple federations have been formed, each comprising different subsets of connectors across the three data spaces. An overview of these federations and the participating connectors is provided in Figure 16.

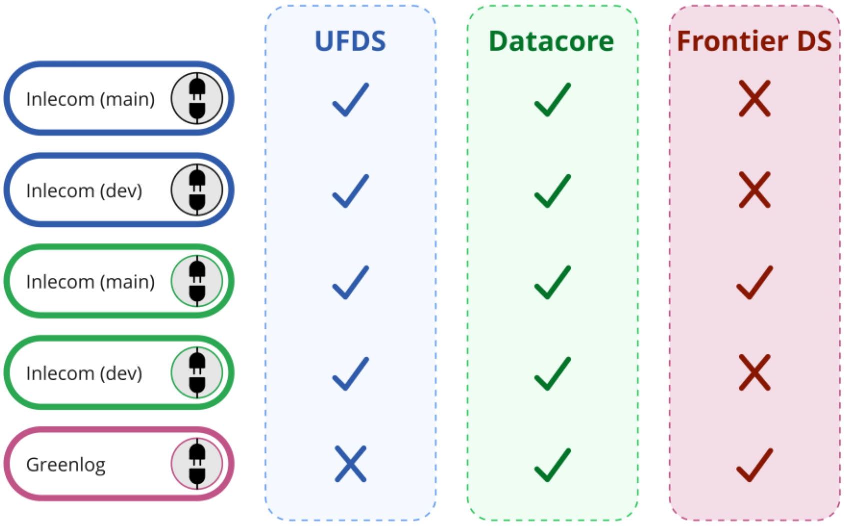

Figure 16: Participation in Federated Data Spaces

Finally, a comprehensive overview of all deployed connectors, including their DSP endpoints and the number of datasets they expose, is presented in Figure 17.

DISCO FEDERATED DATA SPACE DEPLOYMENT			
#	**Connector**	**Data Space**	**Datasets**
1	Inlecom (main)	UFDS	4
2	Barcelona	UFDS	3
3	Ghent	UFDS	3
4	Copenhagen	UFDS	7
5	Thessaloniki	UFDS	4
6	Helsinki	UFDS	6
7	Zaragoza	UFDS	4
8	Valencia	UFDS	2
9	Padua	UFDS	0
10	AGC	UFDS	3
11	Delicode	UFDS	5
12	Technolution	UFDS	3
13	Vianova	UFDS	1
14	CodaQ	UFDS	3
15	CodingTheCurbs	UFDS	3
16	BeMobile	UFDS	2
17	Inlecom (dev)	UFDS	0
18	Akkodis	UFDS	0
19	IMEC	UFDS	1
20	Meta Model Suite	UFDS	0
21	Transportes Callizo	UFDS	2
22	ACS	UFDS	1
23	Next Modular Vehicles	UFDS	1
20	Inlecom (main)	Datacore	1
21	Inlecom (dev)	Datacore	2
22	University of Maribor	Datacore	0
23	Greenlog	Frontier DS	3
		Total Datasets:	**64**

Figure 17: List of deployed connectors and available data assets (the URL of the connectors being publicly reachable are not shown here for security reasons).

4.2. Integration with other Projects

In this section, the integration of the DISCO project and UFDS with other initiatives is presented. Specifically, integrations with URBANE (Section 4.2.1), Green.Dat.AI (Section 4.2.2), and GREENLOG (Section 4.2.3) are described. These integrations have been realised by either involving directly UFDS, as in the cases of URBANE and Green.Dat.AI, or indirectly through the reuse of DISCO's open-source components made available via the Open Software Repository, as demonstrated by GREENLOG.

4.2.1. URBANE

The URBANE project (Grant Agreement No. 101069782) is a Horizon Europe initiative coordinated by INLECOM, focusing on enabling sustainable last-mile logistics and supporting LSPs through advanced digital solutions. A key outcome of the project is the development of the CITIQORE platform, which provides a unified environment for managing and optimising urban logistics operations.

As part of DISCO's effort to expand the number of supported data space applications, the CITIQORE platform was identified as a strong candidate for integration, given its role as a user-facing system for LSPs and last-mile operations. Building on this, three applications were ultimately developed (Section 3.3.6), leveraging CITIQORE as the main user interface. These applications interact with DISCO components for data exchange, primarily the Data Archivist (Section 3.3.3) and the Data Anonymisation (Section 3.3.4). In addition, the Asset Consumer (Section 3.3.2) complements this setup by abstracting data space-specific complexities and enabling straightforward data ingestion. Through this approach, data retrieved from the data space is made readily available to downstream applications via the Data Archivist's REST API, significantly simplifying integration for developers of downstream data space applications.

The Optimal Locker Number Simulator, as the first of the three applications to be developed, was demonstrated during URBANE's final event in Barcelona (November 2025), with a video demonstration available online.[7] The Shareable Lockers Simulator, developed subsequently, is planned to be demonstrated during URBANE's final review (28 April 2026), with a corresponding video having been made available.[9] The third application, the GeoJSON Data Synthesizer and Anonymiser, has been completed from the URBANE side; however, finalisation on the DISCO side requires additional configuration of the Data Anonymisation pipeline, as discussed in Section 3.3.

[9] https://fitconsultingsrl.sharepoint.com/:v:/r/sites/DISCO/Documenti%20condivisi/C.%20EXECUTION%20PHASE/WPs/WP3%20The%20Urban%20Freight%20Data%20Space%20for%20NetZero%20cities/Data%20Space%20Applications/Shareable%20Lockers%20Simulator%20video-demo.mp4?csf=1&web=1&e=y3sL1w

This final application will be fully reported, along with demonstration material, in Deliverable 7.4: Liaison with other initiatives and ICBT.

For further details on URBANE and the CITIQORE platform, the reader is referred to URBANE's Deliverable D3.4: Innovation Transferability Platform Documentation.

4.2.2. Green.Dat.AI

The GREEN.DAT.AI project (Grant Agreement No. 101070416) is a Horizon Europe initiative coordinated by INLECOM, aiming to leverage data spaces and AI to support green and sustainable digital transformation. A key aspect of the project is the development of a data space ecosystem where interoperability with external data spaces is essential to fully realise its impact and vision.

To this end, the GREEN.DAT.AI's Data Space, developed and deployed by ATOS and also based on the Sovity Connector, was identified as an ideal first testing ground for the Federated Data Space Component (Section 3.2.5). To support this effort, a dedicated testing data space was deployed within the UFDS, consisting of two connectors (Inlecom main and Inlecom dev) configured to use an earlier version of the Sovity connector (EDC version 7.2.1) to ensure compatibility with the version used in GREEN.DAT.AI's Data Space. On the GREEN.DAT.AI side, the integration was facilitated through a testing connector deployed by ATOS within their data space for this purpose.

Following the deployment and configuration of the connectors and the Data Space Federation Plugin, interoperability was validated through the exchange of test datasets. Specifically, real-time temperature data for Madrid (provided by GREEN.DAT.AI) and Athens (provided by DISCO), sourced via the Open-Meteo API, were successfully shared across the connectors in the federation.

This integration has been formally documented and published as a white paper available on Zenodo.[5]

4.2.3. GREENLOG

The GREENLOG project (Grant Agreement No. 101069892) is a Horizon Europe initiative focusing on sustainable and efficient logistics operations through data-driven solutions. Although not directly connected to DISCO, GREENLOG is a "sister" project of URBANE and shares complementary objectives in the domain of urban logistics and data utilisation. Within this context, an interoperability scenario was explored whereby GREENLOG would provide anonymised delivery data collected in its Ghent LL, and URBANE would consume this data through the CITIQORE platform to demonstrate analytics capabilities, including the use of the Optimal Locker Number Simulator developed jointly with DISCO (Section 3.3.6).

To enable this interaction, a data space-based approach was selected as the most suitable integration mechanism, given that GREENLOG had developed its own data space infrastructure.

However, as the GREENLOG data space is based on FIWARE connectors, which were not readily deployable within the URBANE environment, an alternative approach was adopted. Leveraging DISCO's open-source components and deployment material, a dedicated data space instance was established using the Sovity Connector stack.

As part of this setup, a specialised data space was deployed for Frontier, consisting of a single connector hosting the anonymised Ghent dataset. This dataset was then made accessible through Inlecom's DataCore data space using the Data Space Federation component, allowing URBANE to consume it within the CITIQORE platform.

Using this setup, a proof-of-concept demonstration was carried out, showcasing the utilisation of the dataset within Citiqore applications. This integration illustrates how DISCO's components and federation capabilities can facilitate interoperability even across heterogeneous data space technologies, enabling practical cross-project data usage scenarios.

5. Documentation and Deployment

All components required for a standard deployment of the data space, including the core infrastructure, as well as the Asset Consumer and Data Archivist applications, are open source and publicly available through the Open Software Repository.[1] Each component is accompanied by comprehensive documentation in the form of README.md files within their respective repositories, ensuring that both technical and non-technical users can easily access guidance on configuration, usage, and maintenance. In addition, the repository includes a dedicated UFDS Platform repository group, which provides structured instructions and supporting software for two distinct roles: Data Space Authorities and Data Space Participants.

For Data Space Authorities, a streamlined deployment process is provided to facilitate the initial setup of the infrastructure. This process is centred around a setup script (setup-deployment.sh), which interactively collects key configuration parameters, such as the domain names under which the various components will be exposed. Based on these inputs, the script automatically generates the required configuration variables, produces secure credentials for all services, and creates a pre-configured NGINX template tailored to the deployment environment. This template can be reviewed and adjusted as needed before being applied to the hosting environment, a step that typically requires minimal involvement from IT personnel. Once this initial configuration is complete, the entire platform can be deployed through a simple Docker Compose execution, requiring only a small number of straightforward commands. All generated credentials are centrally stored in a dedicated vault, allowing administrators to easily access and manage them.

Following the initial deployment, the onboarding of new participants is supported through a dedicated Ansible script (playbook) designed to minimize manual effort. The script requires only two inputs per participant: a unique connector identifier and the hostname under which the participant's connector will be deployed. Using this information, the system automatically provisions the necessary resources, including participant registration in the identity management system (Keycloak), configuration within the Vocabulary Hub, and generation of all required credentials. Additionally, a dedicated folder is created for each participant, containing all necessary deployment artifacts, such as configuration files, credentials, and step-by-step instructions, which can be directly shared with the participant.

From the participant perspective, the deployment process is intentionally lightweight and designed to require minimal technical overhead. Assuming prior onboarding by the Data Space Authority, participants are only required to place the provided files into their local environment, execute a setup script to define basic metadata (such as their organization name and public endpoint), and launch the required services using a single Docker Compose command. A final script handles the automated registration with the identity service, completing the integration into the data space. While some initial configuration of an NGINX server is required to expose the necessary endpoints, a ready-to-use template is provided to guide this process, ensuring that any required involvement from IT personnel remains minimal. Overall, the deployment approach strikes a balance between automation and flexibility, enabling rapid onboarding while maintaining alignment with best practices for secure and scalable infrastructure setup.

6. Moving Forward

The evolution of data spaces must be understood across two interdependent dimensions: the maturation of its technical foundations and the conditions necessary for broad, sustained adoption. Progress along both axes is essential, and neither alone is sufficient.

Converging on a Common Technical Baseline

On the technical side, a key direction concerns the alignment of the platform's core components with emerging European frameworks, particularly the transition towards Simpl and its open-source ecosystem, Simpl-Open.[10] Central to this transition is the progressive convergence of the platform's current implementation towards EDC, which serves as a foundational element of the Simpl architecture. This shift is technically feasible for UFDS (the existing implementation is already based on the Sovity Connector, itself a fork of EDC) and is also strategically necessary, as changes in Sovity's licensing model have introduced constraints that may limit its suitability for fully open and extensible deployments. By consolidating around EDC as a shared, openly governed baseline, the

[10] https://simpl-programme.ec.europa.eu/dashboard/simpl-open

broader ecosystem gains a more unified and interoperable foundation, while reducing the risk of fragmentation that arises from multiple parallel and incompatible implementations.

Importantly, this transition does not render the work developed within this project obsolete. On the contrary, the existing components provide a solid basis for refinement and integration into future architectures. The identity management component, for instance, could be further developed to support decentralised identity paradigms and digital wallets, aligning with emerging trends in self-sovereign identity. Similarly, the logging and auditing mechanisms could be enhanced through integration with distributed ledger technologies, enabling immutable and verifiable record-keeping across participants, whether through selected nodes or a fully federated ledger, ensuring transparency and trust without centralised control. In this sense, the work carried out here constitutes a valuable stepping stone: contributing practical insights, reusable artefacts, and an evolving set of building blocks to the next generation of data space infrastructures.

From Project Infrastructure to Sector-Wide Ecosystem

Technical readiness, however, is only one part of the picture. The true value of a data space emerges when a critical mass of participants is reached, enabling rich, multi-lateral data exchange that goes well beyond isolated use cases. When only a small number of actors are involved, the perceived benefits may not justify the initial investment and learning curve required to adopt such architectures, making simpler bespoke solutions appear comparatively attractive. Overcoming this threshold requires deliberate and coordinated effort.

In this respect, the public sector has a pivotal role to play, particularly in domains such as urban freight, where there is a structural asymmetry between a diverse set of private operators and a central public authority with regulatory and coordination responsibilities. Policy can act as a catalyst: by encouraging or requiring private participants to engage with public sector services through a data space, an initial critical mass can be established. This approach lowers the effective barrier to entry by transforming initial adoption into a necessary investment, while simultaneously creating conditions under which participants can begin to realise additional value through peer-to-peer interactions. As integration costs become sunk and familiarity grows, participants are likely to extend their use of the data space beyond mandatory interactions, giving rise to a network effect capable of driving exponential growth and long-term sustainability. Over time, this trajectory enables the data space to evolve from a project-driven implementation to a sector-wide digital infrastructure, supporting a wide range of operational, regulatory, and collaborative scenarios across the urban freight ecosystem.

Addressing Trust and Demonstrating Value

A closely related set of challenges concerns the barriers that currently limit engagement from private actors, particularly LSPs. A key concern is trust: for many participants, operational data constitutes a core business asset, and the perceived risk of unintended exposure to competitors remains a significant deterrent. This concern is compounded by a widespread misconception that data spaces entail centralised data pooling, rather than controlled, policy-driven sharing in which each participant retains sovereignty over their data. Addressing this misconception through education and transparent communication is essential.

At the same time, technical mechanisms can reinforce confidence. Robust anonymisation techniques, such as those explored through the Data Space Anonymisation application, reduce the risks associated with sharing sensitive information, while frameworks for controlled data monetisation can provide tangible incentives, allowing participants to extract value from selectively sharing subsets of their data under clearly defined conditions. Together, these capabilities transform the data space from a potential liability into an asset.

Even where trust barriers are addressed, a further obstacle persists: the limited immediate return on investment perceived by private actors. When compelling, value-generating applications are absent, even technically capable participants lack strong motivation to engage. The development of targeted applications that directly address the operational needs of LSPs and demonstrate clear, short-term benefits is therefore essential. The public sector can play a significant role here by collaborating with private stakeholders to co-develop such applications, ensuring alignment with real-world requirements; and encouragingly, initial steps in this direction have already been taken, with first proof-of-concept applications now in development as in the DISCO project. Positioning these applications as services that are primarily or exclusively accessible through the data space further strengthens the incentive to onboard. By creating well-designed entry-point use cases that deliver immediate value, it becomes possible to build engagement incrementally, fostering a broader ecosystem in which more advanced and collaborative data-sharing scenarios can naturally emerge.

A Coherent Path Forward

Taken together, these directions point towards a coherent and mutually reinforcing path: a technically robust and openly governed platform, aligned with European standards; a policy environment that supports early adoption and sustains momentum; and a growing portfolio of practical applications that make participation worthwhile from the outset. Each element strengthens the others. As the technology matures, trust increases; as participation grows, applications become more valuable; as applications deliver results, adoption accelerates. The architecture developed within this project, and the lessons drawn from its deployment, provide a concrete foundation on which this vision can be built.

7. Conclusion

This deliverable has presented the Urban Freight Data Space in its final form: a fully deployed, standards-aligned infrastructure for controlled data sharing in urban freight, built over the course of the DISCO project and refined through real-world operation, cross-project collaboration, and iterative development.

The result is more than a collection of software components. The UFDS demonstrates that a principled, interoperable data space for urban freight is technically achievable, deployable at meaningful scale, and extensible to other projects and ecosystems. The integrations with URBANE, Green.Dat.AI, and GREENLOG, alongside the development of the first data space applications, are evidence that the infrastructure can support genuine, value-generating use.

At the same time, the work carried out here is best understood as a foundation rather than a finished product. The path from a project-driven implementation to a self-sustaining sector infrastructure requires continued effort; technically, institutionally, and in terms of the applications that make participation worthwhile. The DISCO project has laid the groundwork. The next step is to build on it.

Funded by the European Union

CIVITAS Sustainable and smart mobility for all

THE CIVITAS INITIATIVE IS CO-FUNDED BY THE EUROPEAN UNION

DISCO is a project under the CIVITAS Initiative.
Read more - civitas.eu

DISCO is a project under the CIVITAS Initiative.
Read more - civitas.eu

