Deliverable 31.1
Federated data space “Sandbox Environment” description

Project acronym:	FP1 MOTIONAL
Starting date:	2022-12-01
Duration (in months):	46
Call (part) identifier:	HORIZON-ER-JU-2022-01
Grant agreement no:	101101973
Due date of deliverable:	2023-06-30
Actual submission date:	2023-06-30
Code	FP1-WP31-D-FSI-001-06
Responsible/Author:	Simone Salviati (FS)
Dissemination level:	PU
Status:	Issued

Reviewed: Yes
Reviewers: Elena Garcia Jiménez (ETRA I+D)
Pietro Pace (MERMEC)

This project has received funding from the European Union’s Horizon Europe research and innovation programme under Grant Agreement No 101101973.

Document history		
Revision	*Date*	*Description*
1		First issue
2	April 20 2023	Second issue
3	May 29 2023	Third issue
4	June 22 2023	Final Draft

Report contributors		
Name	Beneficiary Short Name	Details of contribution
Simone Salviati	FS	First draft
Simone Salviati	FS	Second Draft
Simone Salviati	FS	Third Draft
Simone Salviati	FS	Final Draft

Table of Contents

List of Figures

1. Executive Summary

As stated in its Master Plan, the Europe's Rail Joint Undertaking (ERJU) public-private partnership "aims to accelerate research and development in innovative technologies and operational solutions supporting the fulfilment of European Union policies and objectives relevant for the railway sector and supporting the competitiveness of the rail sector and the European rail supply industry"; [it] "will foster a close cooperation and ensure coordination with related European, national and international research, innovation deployment and investment activities in the rail sector and beyond, in particular under Horizon Europe, Connecting Europe Facility, and the Digital Europe Programme [1].

The creation of a Rail Data Space (RDS) as one of the implementation deliverables of the ERJU[2] is a concrete instance of cooperation and coordination with the Digital Europe Program: its ultimate goal is to align data sharing and communication in the Rail Sector to the European Data Strategy, making the Rail Sector both a contributor and a beneficiary of the "single market for data" for a "data-driven" European society[3]:

- Leveraging the resources and know-how provided by the Dataspace Support Center[4,] funded by the Digital Europe Programme,
- Ensuring at inception and by design compliance with the European legislative acts that complement the establishment of European data spaces (Data Actions)
- Ensuring at inception and by design the ability to deploy the Rail Data Space on Next Generation Cloud infrastructure (Cloud Actions)

[1] Europe's Rail Master Plan: https://rail-research.europa.eu/wp-content/uploads/2022/03/EURAIL_Master-Plan.pdf
[2] ivi, chapter 4.2.2.6 "Transversal topics: data and digital enablers"
[3] European Data Strategy: https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/europe-fit-digital-age/european-data-strategy_en
[4] Dataspace Support Center: https://dssc.eu/

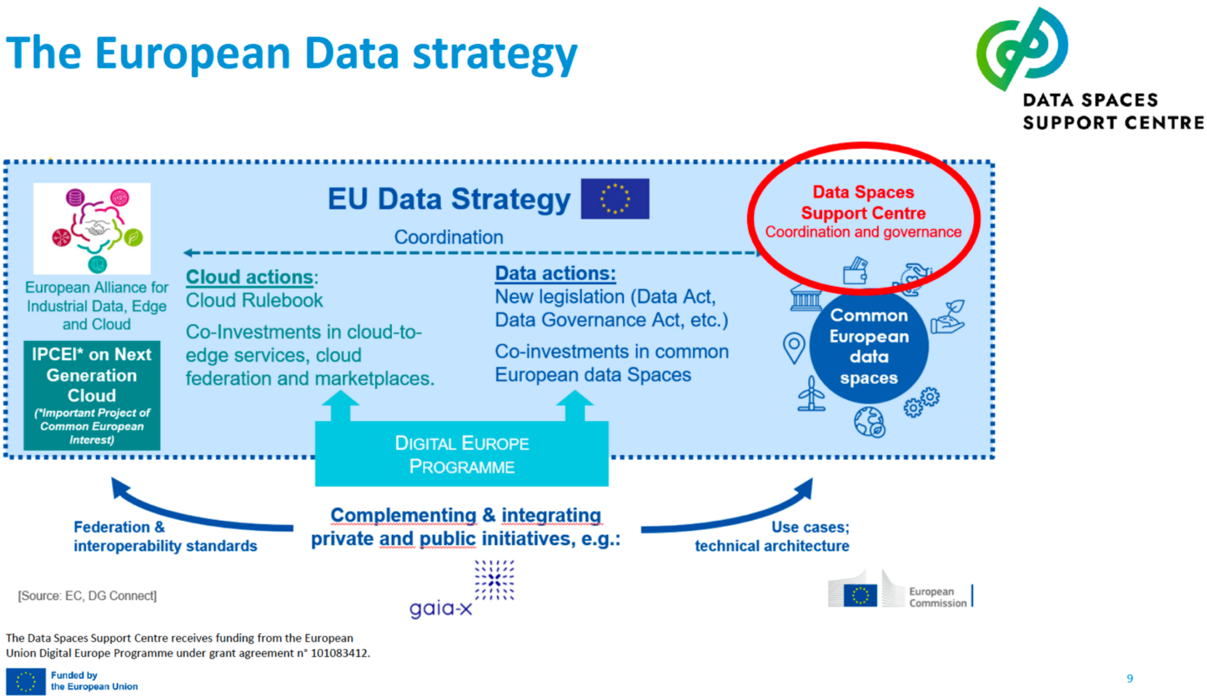

Figure 1 - Data and Cloud Actions, Dataspaces and DSSC in the European Data Strategy

However, the MOTIONAL project is tasked with delivering a viable, usable Rail Data Space as the implementation of a common enabler for data sharing and communication for all ERJU flagship projects and relevant elements of the System Pillar designed rail system architecture. This circumstance adds an essential timing constraint to its delivery, which must be compatible with the roadmaps and execution plans of its intended users, e.g. Rail Undertakings, Infrastructure Managers and Rail Industry participants in all other flagship projects.

The “sandbox environment” described in this document is designed to provide acceleration to the process of delivering a viable, practical and usable Rail Data Space to partners in the Innovation Pillar that meets their timing constraints while, at the same time, ensuring alignment ‘by design’ with European Data Strategy goals and with future releases of additional technology features, components and services developed by organizations, described in section 4.1, operating with support from European Institutions on their own timelines.

2. Abbreviations and acronyms

Abbreviation / Acronym	Description
DSBA	Dataspace Business Alliance
DSP	Data Spaces Protocol
DSSC	Dataspace Support Center
EDC	Eclipse Dataspace Components
ERJU	Europe’s Rail Joint Undertaking
FCN	Federated Catalog Node
FCC	Federated Catalog Crawler
FDS	Federated Data Space
GAIA-X	GAIA-X Association
IDSA	International Data Space Association
RDS	Rial Data Space

3. Background

The present document constitutes the Deliverable D31.1 “Federated data space ‘Sandbox - Environment’ description” in the framework of the MOTIONAL Flagship Project as described in the project’s Grant Agreement (101101973 – [FP1 MOTIONAL]).
It is part of Work Package 31 “Federated Data Space”. As the MOTIONAL project progresses, this document will be updated in order to include any changes/deviations and or needed modifications until the end of the project.

4. Objective

The WP31 aims to deliver a trusted, reliable, cybersecure federated data space for the rail ecosystem - the Rail Data Space. The Rail Data Space will provide exchange and sharing of digital resources across Rail operators, Infrastructure Managers and Suppliers as a contribution towards building the European Mobility Data Space.

Common European Dataspaces[5] are a key element of the overall European Data Strategy, which is supported by European Institutions and large or specialist private and public Companies. While the strategy is clear, European funding is available in different programmes, numerous real world 'building block' components and concrete dataspace implementations are being developed at the time of this writing, the endeavour is nonetheless ambitious, the roadmap relatively long, and - finally - the landscape of base system technologies and of participants is complex, as discussed below.

The MOTIONAL project must however execute under a defined timeline, and a viable, ready-to-run Rail Data Space in particular must be delivered, as a common enabler, for other Innovation Pillar Flagship Projects in time for its exploitation.

In order to meet the twin requirements of delivery in time to Flagship Projects and consistency 'by design' with the European Data Strategy, the MOTIONAL project had deliberately chosen a strategy for development in which existing 'building blocks' already available from current large scale data space projects are assembled in a 'sandbox' for further development.

The sandbox is an environment that closely mimics the real-world scenario, needed to support developers in design, development, testing, validation and deployment operations of the Rail Data Space.

This approach is designed for the following purposes:

- Start development from proven software components that are consistent with the data space principles adopted and maintained by organizations supported by European Institutions. This is at the same time a measure of *acceleration* of the development process, an approach to maintain *consistency* with common European dataspace principles, and a means to *validate* concepts, technologies and development *by running code*. This is an important approach to ensure later interoperability of the rail data space with other data spaces of similar architectures (cross-dataspace compatibility).
- Contribute to *standardization* with specifications *validated* and *proven* by actual *running code,* particularly of *interoperability* across dataspaces, e.g., with the Mobility Data Space currently developed as a Lighthouse project of the GAIA-X Association (see below).

[5] A definition of the principles, concepts and architecture of a data space is beyond the scope of this document and can be found in the literature listed in its bibliography).

This document provides a technical description of the federated data space "Sandbox-Environment" to be used by project participants for development and testing of the Rail Data space software components to be delivered by the MOTIONAL Work Package 31.

Section 6 "Definition" describes the Sandbox Environment components and system requirements.

Section 7 "Procurement" describes the process of sourcing, installing and configuring the Sandbox Environment in order to match its definition in a concrete installation ready to be used by Rail Data Space developers. In addition, it describes the installation and configuration in such a way that it could be replicated autonomously as needed in a different hosting computer environment.

Section 8 "Delivery" describes the process of making the Sandbox Environment installation available to Rail Data Space developers in the course of software development and testing. In addition, it describes the process of keeping the installation viable and up-to date as development progresses and/or additional developers are added to the development team.

4.1. The current European Dataspace landscape

The implementation of the European Data Strategy is supported by a number of Institutions, Associations and Companies. These include:

- The Dataspace Support Center, funded by the Digital Programme Europe
- The Dataspace Business Alliance[6]
- The GAIA-X European Association[7]

The figure below provides a depiction of technology aspects of the European Strategic Digital, Data and Dataspace initiatives:

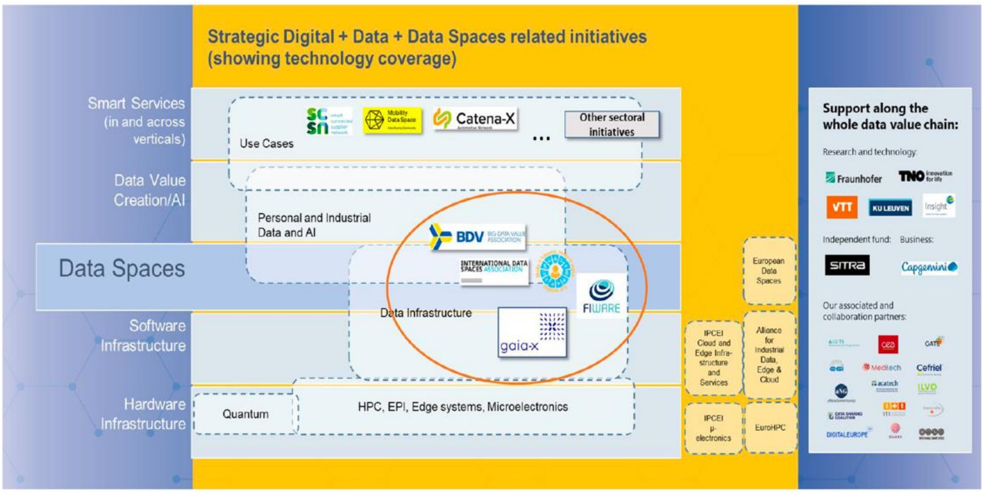

Figure 2 - Data space initiatives - technology coverage

[6] Dataspace Business Alliance: https://data-spaces-business-alliance.eu/
[7] GAIA-X European Association: https://gaia-x.eu/who-we-are/association/

GAIA-X is also a consortium member of the Dataspace Support Center and a member of the Dataspace Business Alliance. It is organized into a:

- Data Spaces Business Committee concentrating on Operational Requirements of a production dataspace
- Policy and Rules Committee working on Policy rules and the Trust framework for participation in a dataspace
- Technical Committee working on the Federation services and Architecture of a dataspace, based on the International Dataspace Associations[8] "Reference Architecture Model".

A number of sectorial dataspace "Lighthouse projects" are in actual development within the GAIA-X Association, including:

- Agdatahub (Agriculture)[9]
- Catena-X (Automotive Supply Chain)[10]
- EONA-X (Mobility, Transport and Tourism)[11]
- EuProGigant (Manufacturing, Industry 4.0)[12]
- Mobility Data Space (Mobility)[13]
- SCSN (Electronics Supply Chain)[14]
- Omega-X (Energy)[15]
- GAIA-X4 Future Mobility (Mobility and Transport)[16]

While not formally a GAIA-X Lighthouse project, the Rail Data Space to be developed in the MOTIONAL project is nonetheless effectively positioned as one of the sectorial initiatives on the top of Figure 2, and as such it can leverage the common 'technology coverage' described in the red circle of the same figure, available to all European Dataspace initiatives. This coverage includes GAIA-X-delivered guidelines, open-source 'bulding blocks' software and the GAIA-X Digital Clearing House services[17] that support the implementation of the GAIA-X Trust Framework, which is common and mandatory for all European data spaces . This is necessary to guarantee formal compliance with GAIA-X, and therefore European, governance rules for participation in dataspaces, as well as compatibility with new components, features and services under development and interoperability within and across dataspaces.
The MOTIONAL WP31 Sandbox environment will therefore be based on the same set of 'building blocks' and services provided by GAIA-X / IDSA through the Dataspace Support Center.

---

[8] International Dataspace Association: https://internationaldataspaces.org/, a member of GAIA-X
[9] https://agdatahub.eu/en/
[10] https://catena-x.net/en
[11] https://eona-x.eu/
[12] https://euprogigant.com/en/
[13] https://mobility-dataspace.eu/
[14] https://smart-connected.nl/en
[15] https://omega-x.eu/
[16] https://www.gaia-x4futuremobility.de/
[17] GAIA-X Digital Clearing House: https://gaia-x.eu/gxdch/

4.2. A developing *market* of Dataspace providers

As an element of the European Data Strategy funded and facilitated by European Institutions and implemented by Associations including small and large technology partners and vendors who develop standard specifications for architecture and services, dataspaces are attracting investment from Commercial suppliers and service providers, with some major Consulting Companies, among which many members of the GAIA-X association, establishing business units and practices dedicated to them: a market for dataspace technology, components and services for establishing dataspaces is developing including Deutsche Telekom/T-Systems, SAP, Capgemini, Accenture, ATOS and KPMG.

This circumstance shall be further investigated and leveraged with the goal of evaluating different strategies for the further refinement of the definition and especially for procurement of the sand-box environment. In addition, this factor will pay a role in the establishing a roadmap and strategy for promoting the sandbox to a production environment operated by a professional dataspace operator.

5. Federated Dataspace concept overview

The cybersecure, reliable, scalable, interoperable, vendor-independent and technology-agnostic exchange of digital resources among trusted partners is a recurring problem everywhere digitalization opens up opportunities for automation, optimization, and the creation of new products and services for society. Finding a recurring, repeatable solution for such a problem is a critical necessity to avoid throttling down the pace of the digitalization process.

While numerous schemes and technologies exist for performing the actual exchange of digital resources, the persistent issue limiting their applicability to relatively small domains is the lack of appropriate provisions enabling the digital definition and the digital enforcing of policies that ensure the identity of the parties involved in the exchange, and allow the owners of the exchanged digital resources, in the production of which large investments are made, to establish and enforce ownership rights and control on their access and usage. As for any valuable asset, a market of digital products and services cannot be established unless the participants in a transaction are not guaranteed 'property' rights, unless they can freely negotiate 'contracts', and unless they are subject to the obligations they assume under the contract and applicable laws and regulation, including responsibility for the quality and the usage of the assets.

A Dataspace can be considered as an architectural pattern, i.e., "a named collection of architectural design decisions that are applicable to a recurring design problem, parametrized to account for different software development contexts in which that problem occurs"[18]. As described above, the specific problem at hand extends beyond the mere exchange of data but covers the issues associated with data ownership and control, identity, trust and business autonomy of dataspace participants.

A simplified view of the Dataspace concept is provided in the following figure:

[18] Taylor, R.N.; Medvidović, N.; Dashofy, E.M. (2009). Software architecture: Foundations, Theory and Practice. Wiley.

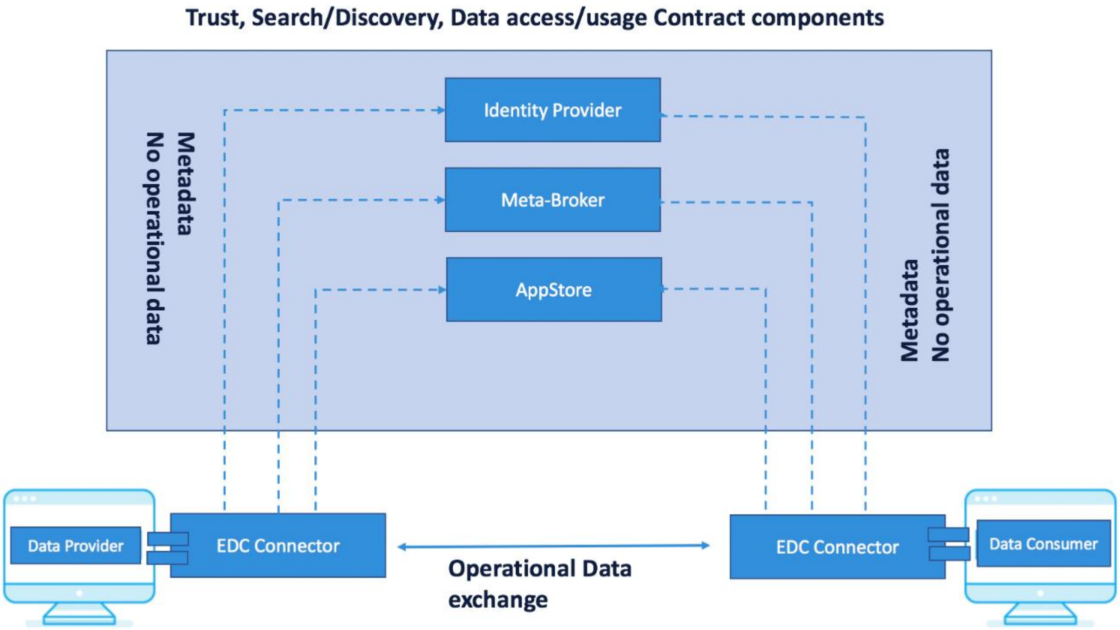

Figure 3 - Essential elements of a Dataspace

The essential features of the pattern can be summarized as follows:

- The Dataspace is a *community* of autonomous Participants who play the roles of Data Provider and Data Consumers. They become Participants by identifying themselves to the community through the Identity Provider services and implementing/installing an "EDC (Eclipse Dataspace Components) Connector" (described in section 6.5 of this document) registered with the Dataspace through the Meta-Broker. Becoming a Participant or leaving the Dataspace is an autonomous decision of the Entity.
- The Dataspace common components provide trust and identity, search and discovery, and contract negotiation and enforcement (web) services. They do not collect, store or forward the actual digital assets, and they do not perform or are involved in the actual digital assets exchange.
- Actual data exchange is peer-to-peer between the mutually recognized participants in the mutually contracted exchange through a mutually agreed protocol implemented by the EDC Connector. Data assets are stored *at the Participant nodes*.
- The architectural pattern is *distributed* and 'parametrized' in the sense that it can be implemented and deployed on any suitable computing and system software environment: other than being able to run a computing and operating-system agnostic open-source EDC Connector, a Participant has no constraint on technology stacks or programming languages at its end of the exchange.

Since implementing and executing an EDC Connector is the only technical requirement for an Entity to be a Participant in a Dataspace, the Entity can be a Participant of multiple sectorial dataspaces at the same time, e.g., Mobility Dataspace, Energy Dataspace, etc. In fact, a federation of sectorial dataspaces is possible and envisioned by the European Data Strategy and the GAIA-X Association, as depicted in the following figure:

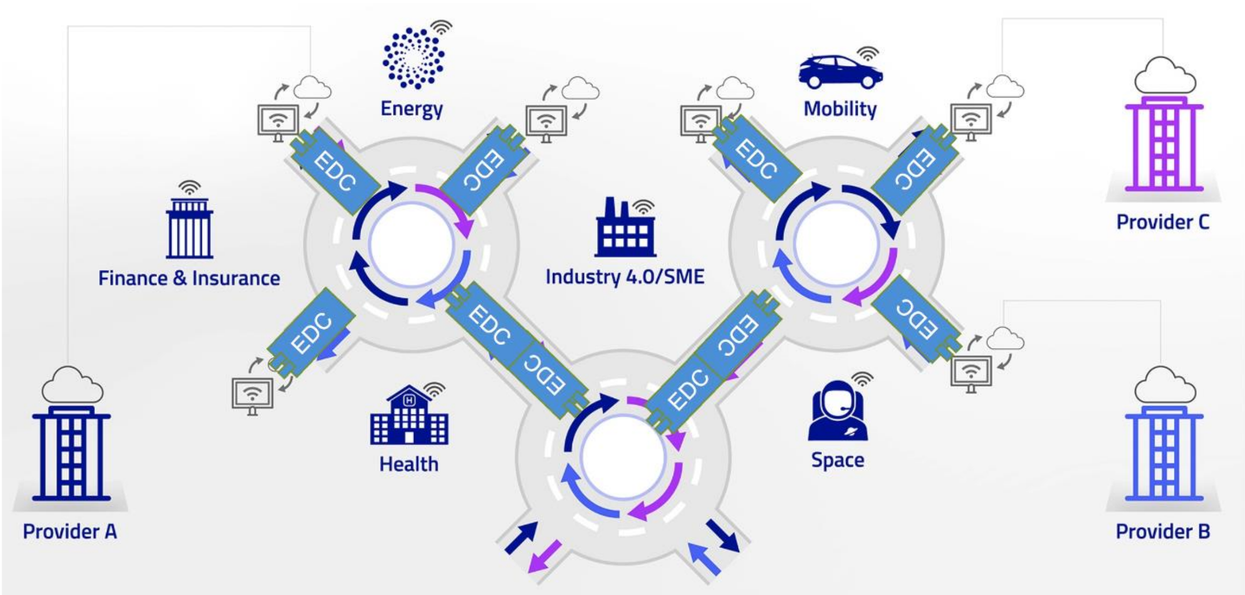

Figure 4 - Federated Dataspaces

This is achieved through federation protocols that synchronize the individual Dataspace common components, i.e., Meta-broker, Identity Provider. This also means that individual, usually large, organization could create dataspaces 'private' to the organization that use the pattern to share data asset across the Organization's constituent Affiliated organizations, and then federate the 'private' dataspace with other dataspaces of one or more sectors.

Details on the design and architectural principles of dataspaces are available in the specialist documentation:

- GAIA-X Technical architecture[19]
- IDSA Reference Architecture Model[20]
- Design Principles for Dataspaces – Position Paper[21]

[19] https://www.data-infrastructure.eu/GAIAX/Redaktion/EN/Publications/gaia-x-technical-architecture.html
[20] https://github.com/International-Data-Spaces-Association/IDS-RAM_4_0/
[21] https://design-principles-for-data-spaces.org/

6. Federated Dataspace "Sandbox Environment" Definition – Blueprint

6.1. Introduction

WP31 aims to deliver a trusted, reliable, cybersecure federated data space for the rail ecosystem - the Rail Data Space.

To facilitate the development of WP31 components/artifacts, it is essential to provide a development environment that is pre-installed with the minimum required components. Task 31.1 aims to define, procure, and deliver a "Sandbox-Environment" to support development of the federated data space for the rail ecosystem. This will allow team members to develop, extend, debug, and release the components as they are rolled out. The availability of such an environment will significantly enhance the efficiency and productivity of the development teams, allowing them to focus on creating high-quality components without the need for time-consuming and tedious setup processes. In this section/chapter, we will discuss what we mean by sandbox, the importance of a pre-installed "Sandbox-Environment" and how it can benefit the project team giving a detailed description of each technical component and the hardware and software requirements needed to deliver it.

6.2. Definition of "Sandbox"

A sandbox is a developing and testing environment where developers can create and test their code close to a production scenario without running the risk to break a actually deployed production environment. It is a safe and controlled 'playground' environment where developers can run software without the risk of damaging or interfering with the live or production system. A sandbox typically provides a replica of the production environment, allowing developers to test their code in a near-real-world scenario. The benefits of using a sandbox include reduced risk, enhanced software quality, and quicker software delivery.

The primary importance of using a sandbox is to minimize risk. In a production environment, even the slightest mistake can cause significant damage to the system. If a developer makes an error in the code, it could result in system failure or data loss. Such a data loss can still occur using a sandbox but the consequences do not propagate and affect the sandbox environment only.

Another critical benefit of using a sandbox is the potential to improve software quality of components. Developers can use the sandbox environment to test components, detect bugs and fix them before they are released to the production environment. This reduces the risk of failures in the production environment. With this testing approach developers can start development, experiment with different frameworks and solutions, and ramp up to maturity without running into the risk of causing damage

6.3. The Rail Dataspace Sandbox

The sandbox environment for the development of the Rail Dataspace in the MOTIONAL project must provide a "minimum viable dataspace", i.e., a minimum set of components and functionalities required to provide a basic data exchange capability. Its purpose is to quickly establish a working Data Space that can be used to develop required extensions for the realization of data asset exchanges implied by the Europe's Rail Joint Undertaking use cases and demonstrations. To procure and deliver the sandbox environment quickly, and in order to tap into an existing supply of specialist professionals already familiar with the technology, the sandbox environment will be established using the Eclipse Dataspace components described in section 6.4 below, i.e., set of open-source frameworks and software tools already in use by developers of numerous sectorial dataspaces and available 'out-of-the-box' as a service from industrial vendors. The following sections describe the Eclipse Dataspace components and the process for establishing the "sandbox" minimum viable dataspace.

6.4. Eclipse Dataspace components

The reference architecture and design principles described above do not imply a specific implementation. However, most sectorial dataspaces under development, such as the GAIA-X Lighthouse projects, are based on open-source software components, the "Eclipse Dataspace Components", developed by a group of European companies under the Eclipse Foundation governance. These components have integrated the GAIA-X Trust Framework and are therefore compliant with GAIA-X defined governance rules for participation in a European dataspace.
In the MOTIONAL project the Rail Dataspace will be created using these components installed in an extended sandbox, described in section 7 of this document, accessing the GAIA-X Digital Clearing House (GXDCH) which provide digitalized services to validate and enforce GAIA-X compliance of the dataspace.

The Eclipse Dataspace components implementation context can be expressed by the following context model:

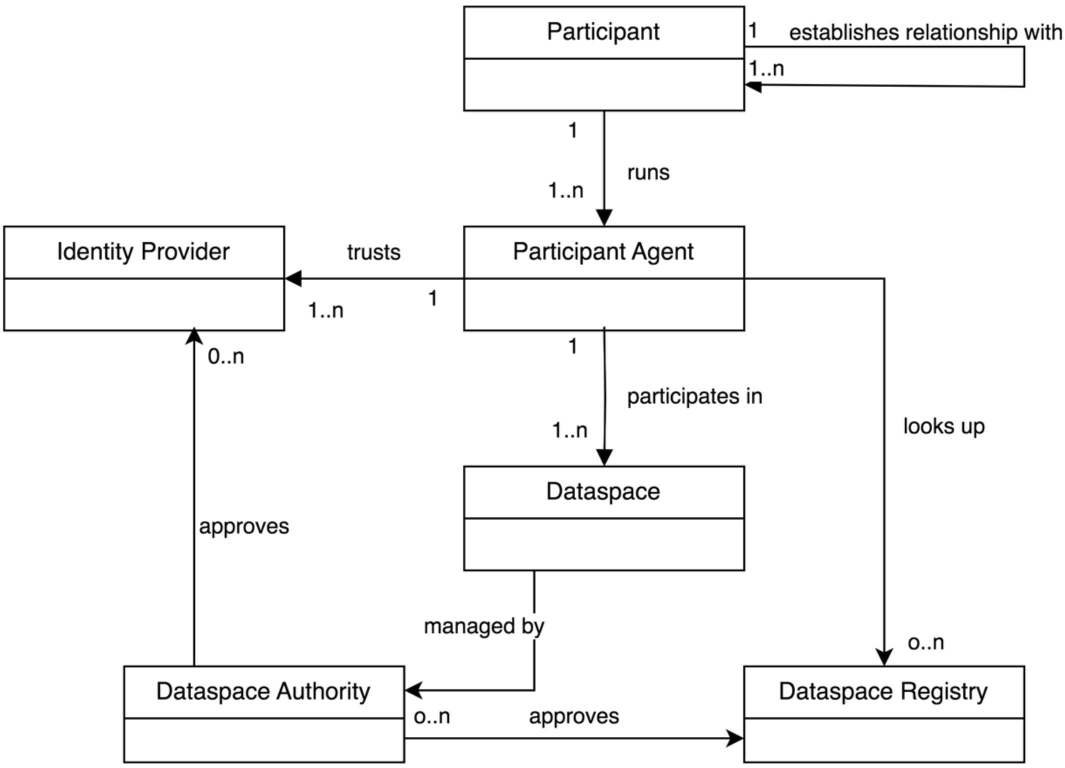

Figure 5 – Eclipse Dataspace context

Participant agents are software systems that perform a specific operation or role in a dataspace. The following illustrates the different types of participant agents that may exist in a dataspace:

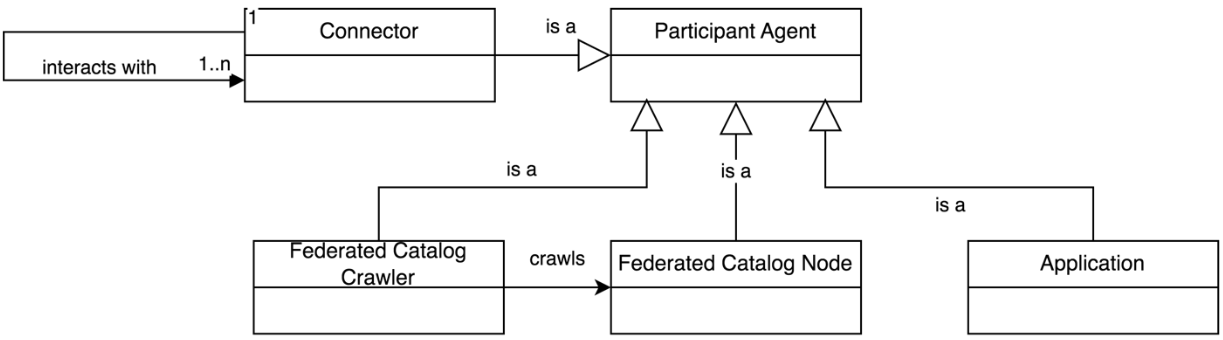

Figure 6 - Participant Agents

The participant agent types are:

Federated Catalog Node (Meta-Broker): A system that publishes a metadata (digital) *description* of the assets, not the actual assets, provided by a participant in a dataspace. Publishing makes the assets available for discovery. A participant may choose to make an asset available to a subset of other participants using an *access policy* and impose usage requirements with a *usage policy*.

Federated Catalog Crawler: A system that discovers metadata descriptions of the assets published by other participants in a dataspace. The result of a crawling operation is a collection of assets the crawling participant has access to. Access is determined by the provider participant and may include evaluation of access policy and usage policy against a set of verifiable credentials. The crawler is used by Data Consumers to discover data assets made available by Data Providers.

Connector: A system that performs contract negotiation and asset sharing (data transfer or compute-to-data) on behalf of a participant. This component is further described in section 6.5 below.

Application: A custom system that performs some role in the dataspace. Applications are the end-users of Data Consumers who operate on the exchanged data assets, such as Traffic Management or Maintenance. Applications are shielded from the data sharing mechanism, and conversely the data sharing mechanism is independent from and common to all applications.

In Figure 5 (above), the notion of a Dataspace Authority is introduced. The Dataspace Authority is responsible for approving one or more identity providers that serve as trust anchors in a dataspace. The Dataspace Authority is an optional role; a dataspace may exist where there is no central authority, or the central authority is implied or enforced 'locally', such as in the 'private' dataspace of an Organization, or it is composed of autonomous actors with no centralized decision-making process.

There is, however, at least one *Identity Provider* associated with a dataspace since all *participant agents* must be identifiable. This is an important distinction: while a participant organization has an identity, all participant agents also have a unique identity. Furthermore, the participant agent identity may be hierarchically related to the participant *organization* identity, thereby making it possible to establish a trust chains. Consider the following scenario, which underscores why this distinction is important. Company A may have two participant agents located in different geographic regions. Based on their location, one participant agent may access geospatially restricted data the other agent cannot access. Access policy would be determined using verifiable credentials tied to the participant agent identity.

An identity provider may be centralized, distributed, or a combination of the two. In the example below a distributed scheme is shown in which two identity providers validate one Participant each. Participant Agents, e.g., Connectors, tun by each Participant establish a trust relationship with both Identity Providers.

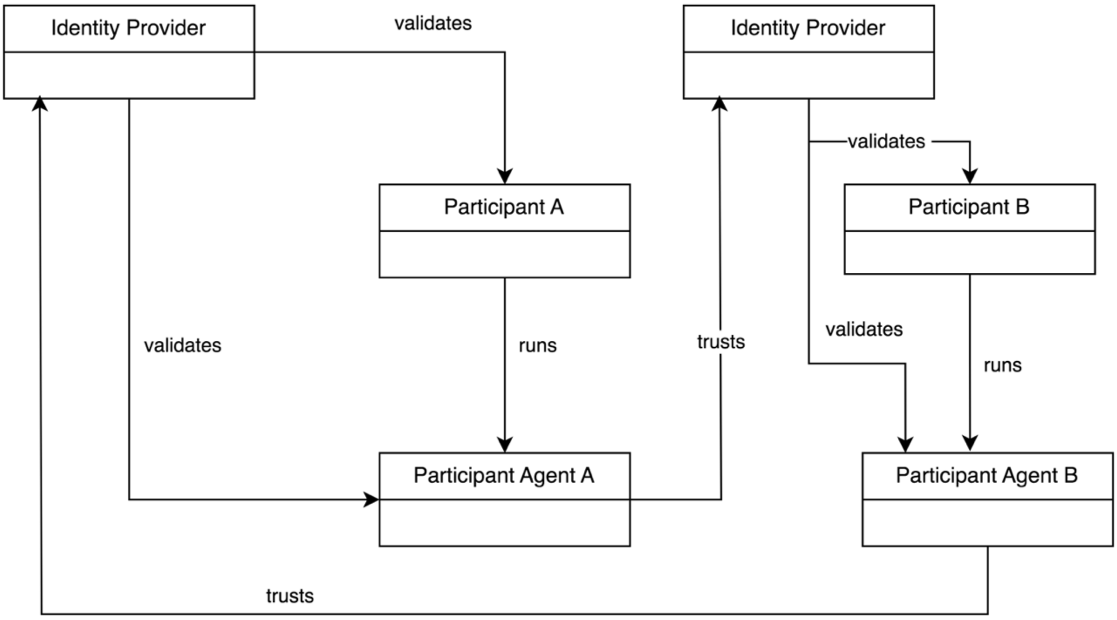

Figure 7 - Identity and Trust in the dataspace

In a dataspace with one centralized identity provider, both Participant A and Participant B would share the same provider.

The Dataspace Authority for European Dataspaces will eventually become a European Institution, or an Organization delegated by it, providing digital "notary services" accessed by the Gaia-X Digital Clearing House, which acts as the mandatory Identity Provider.

There are two types of Catalog Participant Agent: The Federated Catalog Node (FCN) (Meta-Broker) and the Federated Catalog Crawler (FCC). The FCN is used to publish assets to a dataspace. The details of publishing are described in the following section on contract negotiation. It is important to note that the EDC-based FCN is not an asset repository in the classic sense; it does not store data assets. Rather, it is an index or register of assets and pointers to content stored in diverse systems owned and managed by Dataspace Participants. The role of the FCN is to make that index available for discovery by other participants.
The FCC is a participant agent that queries (or crawls) other FCNs in a dataspace. It may be required to present verifiable credentials used to determine which assets are visible to it. A naïve implementation of an FCC could perform real-time crawling in response to a query made by an end-user. This would not scale for dataspaces of any significant size. The EDC FCC, in contrast, performs periodic crawling operations of other FCNs and updates a local, query-able cache. The following diagram illustrates the relationship between the FCC and FCN:

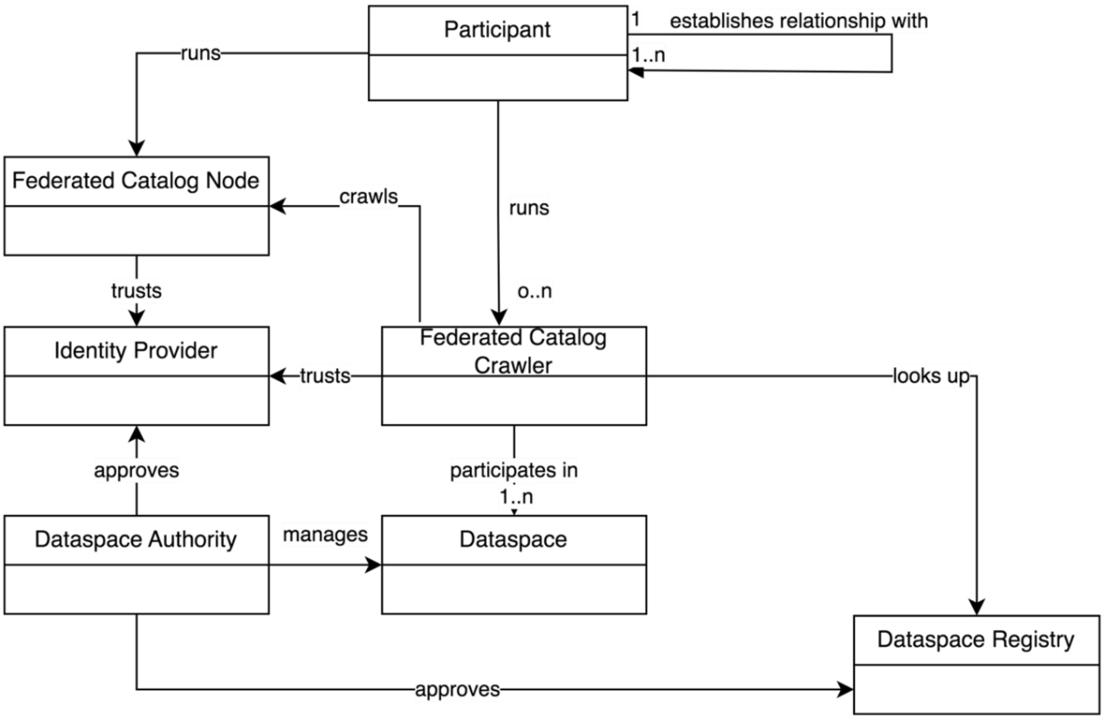

Figure 8- Federated Catalog (Meta-broker) and Crawler

A Connector is a specialized participant agent that functions as the asset sharing infrastructure in a dataspace. It is the embodiment of a Participant *organization* in the Dataspace *community.* Connectors may share diverse assets such as data streams, API access, big data, or compute-to-data services. They may support push data transfers, pull data transfers, event streaming, pub/sub notifications, or a variety of other transfer topologies. The following outlines the role of the connector in a dataspace:

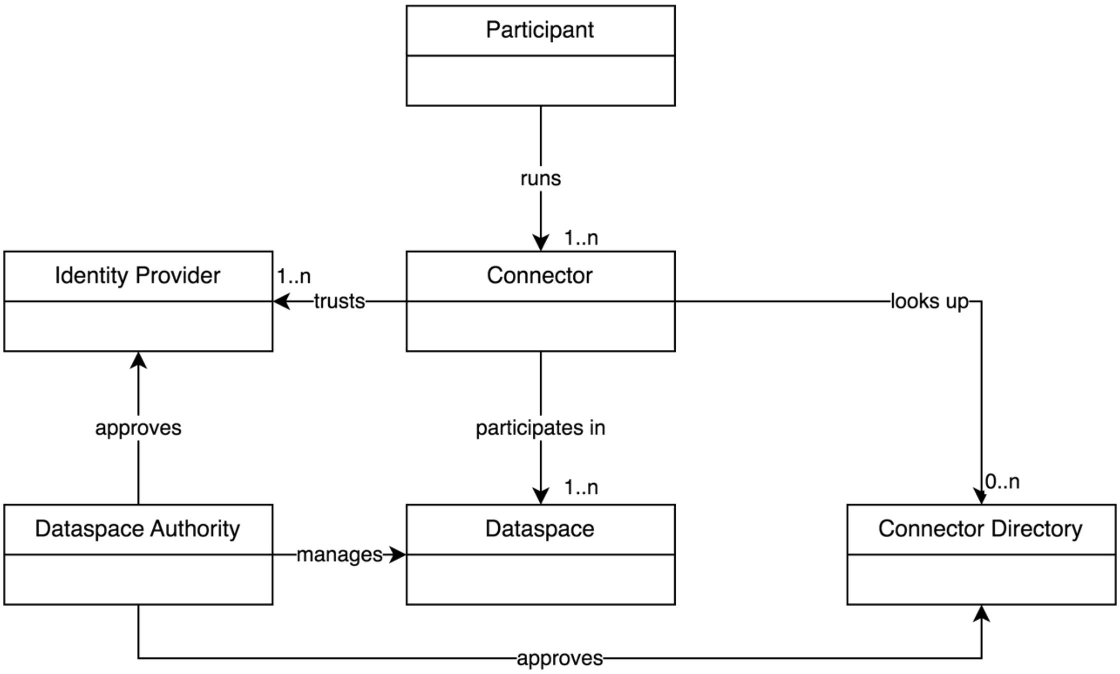

Figure 9 - Connector in the Dataspace

Asset sharing is performed in two distinct steps: *contract negotiation* and *data transfer*. In the EDC, both contract negotiation and data transfer are implemented as asynchronous state machines. Processes transition through a series of states that are understood by the client and provider connectors. Some dataspaces may optimize the contract negotiation step by transitioning it automatically when an asset is requested. Other dataspaces (or, more precisely, participants) may implement a contract negotiation process backed by automated or human workflow. The role of the connector is to manage these processes and provide an audit history of all operations.

6.5. Eclipse Dataspace Connector (EDC Connector)

The Eclipse Dataspace Connector (EDC Connector) provides a framework for sovereign, inter-organizational data exchange, containing modules for performing data query, data exchange, policy enforcement, monitoring and auditing. It implements the IDSA Dataspace Protocol (DSP) as well as relevant protocols associated with GAIA-X.

It is written in pure java code using open-source libraries, is built using the gradle build automation open-source software[22], and can be deployed on any suitable computing and system software runtime environment. It is constituted of:

[22] https://docs.gradle.org/current/userguide/what_is_gradle.html

Core components: all essential building elements that are necessary to run a connector, such as TransferProcessManager, ProvisionManager, DataFlowManager, various model classes, the protocol engine and the policies and contract negotiation management components.

Data Protocols: implementations for communication protocols a connector might use, such as the IDSA Dataspace Protocol; or communication protocols with the Google Cloud Platform, with the Azure Cosmos Platform or with Amazon Web Services.

Launchers: connector packages that are runnable. What modules get included in the build (and thus: what capabilities a connector has) is defined by a specific the gradle build file. They provide a method to control how the connector is launched and becomes operational in a specific computing and system software environment, such as bootstrapping, initialization, managing configurations, allocating resources, etc.

Extensions: components that extend the connector's core functionality with technology- and cloud-specific code.

Service Provider Interface (SPI): the primary extension mechanism for the Connector providing a framework for implementing its interfaces.

The EDC Connector separates the "Control Plane", i.e., the components, services and protocols that interact with the Identity Provider, the Federated Catalog (Meta-Broker) and the services that handle publishing of data asset descriptors and contract negotiation, from the "Data Plane" which performs the actual exchange with another connector. The control plane only exchanges metadata with the common Dataspace services: actual operational data is only exchanged with a verified Participant under a mutually agreed and enforced contract.

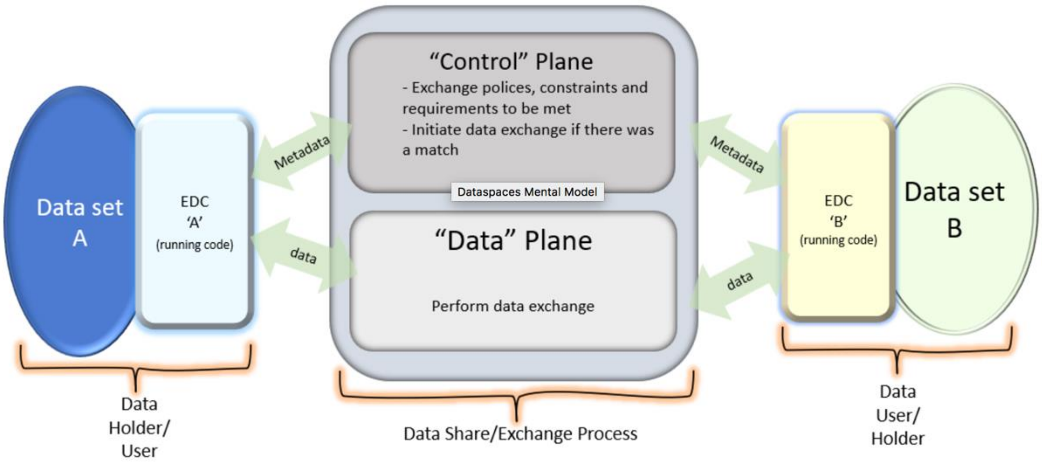

Figure 10 - EDC Connector

An EDC Connector for a specific participant environment, for example integrating legacy systems, accessing local data bases, or running on Microsoft .NET, is built by creating extensions, e.g., a transfer process store based on Azure CosmosDB, SPI implementations, e.g., Connector services called by an existing application, specific data transfer protocols, e.g., through MQTT Queueing, and launchers, e.g., for initialization of the local environment.

Given the underlying architecture of the EDC Connector extensions are *loaded* at run time using the standard java ServiceLoader mechanism , but they must be *defined* at build time. A specific EDC Connector is therefore created by including the extensions in the automated build process that creates the containarizable executable package ready for deployment.

In the scope of the MOTIONAL project, Work Package 31 “Federated Data Space” development tasks will be mainly concerned with building extensions, SPI implementations, data protocols and launchers. The availability of a “Sandbox” with a pre-installation of Eclipse Dataspace components is necessary to concentrate efforts on delivering ‘value-added’ extensions for Rail that enable the implementation of Europe’s Rail Joint Undertaking use cases and demonstrators at the Data Plane level avoiding wasted effort in re-creating the complex Control Plane implementation software.

6.6. Establishing the Sandbox minimum viable dataspace

6.6.1. Assumptions

- Multiple Companies are members of a dataspace; they collaborate on sharing their data with each other. However, they need to retain full control on their digital assets. The dataspace as such must not collect, store or forward the data: only eligible partners specified by the data owners who accept the access and usage policies, also specified by the data owners, can be a party in the exchange of a specific data asset.
- Companies need mechanisms to advertise available data and the policies they want enforced on access and usage of their data.
- Companies must be in control of whom they trust
- Dataspace technology must be compliant with the IDSA/GAIA-X Reference Architecture Mode, vendor-independent, cloud-agnostic, capable of being containerized in packages including the minimum operating system and dependency libraries, able to be ported and executed on any infrastructure. It must impose the minimum set of system requirements to Companies who participate in the dataspace.
- Data exchanges must be interoperable across heterogeneous applications, operating systems and computing hardware.
- Companies must be able to log and trace all interactions with other Companies through the dataspace. Dataspace technology must provide transparency of the interactions for performance assessment and tuning, auditing and quality checks.

6.6.2. Creating the Sandbox Minimum Viable Dataspace

- MOTIONAL Work Package 31 participants act as the Dataspace Authority during development, namely
  - Establish rules of participation.
  - Optionally establish a set of third-party verifier organizations.
  - Define a common set of policies that must be supported by all members.

    - Define a trust model. A trust model may be centralized or decentralized
- The Dataspace Authority defines a registration process
    - Web Application where companies can apply for membership
    - Membership applications are certified by a verifier organization, such as the GAIA-X Digital Clearing House, or an automated ruleset.
    - On verification, the new member company becomes a participant, and an entry is added into the dataspace member registry
- The Dataspace Authority runs the following software components of the Sandbox to perform its role of *manager* of the Dataspace
    - A custom registration web app
    - The EDC Dataspace Member Registry that contains information about the participants and serves as a participant directory
    - One or more Identity Provider(s)

6.6.3. Rail Data Space development using the Sandbox Environment

Acting as a community of Dataspace developers, MOTIONAL WP31 participants

- Clone standard EDC Connectors into a GitHub code repository and version management installation, which is a part of the Sandbox environment, but not of the Dataspace. This step is necessary to make the EDC Connector 'core' components available to the automated build process when including custom extensions developed locally.
- Create pull requests on the GitHub repository and use their own Integrated Development Environment software to create the necessary custom extensions for the EDC Connectors, based on specific data exchange requirements derived from the Europe's Rail Joint Undertaking Use Cases and Demonstrators. These may include service interface provider implementations, data-protocols and launchers as described in section 6.5. Include the custom extensions in the automated build process that packages them with the cloned EDC 'core' components.
- Create pull requests on GitHb to develop auxiliary 'mock' services that emulate behaviors, patterns or functionalities of participant's legacy systems during development in order to drive and validate the dataspace exchange mechanism with realistic scenarios prior to actual integration. The mock services can be divided into two main categories

    - Behavioral services: they emulate, both independently and at the request of the different participants, behaviors and actions of other participants. They support
        - Production of a dataset of different sizes, formats and frequencies
        - Production of model Contract Definitions describing policies on data access and usage
        - Publication to the Federated Catalog through EDC Connector

        - Exchange using different protocols, e.g., synchronous, asynchronous, push, pull, encryption, compression, etc. through EDC Connector
        - Search/Discovery and Consumption by other participants
    - Computational services: they are services that enable data transformations and re-encodings for the different participants:
        - Applying masking features
        - Translation from local dialects into the ontology of the information model
        - Orchestration between different datasets and creation of composite data assets.
- Developers may also need to extend the Federated Catalog (Meta-Broker). In this scenario, the Federated Catalog Eclipse dataspace component is also cloned to the sandbox GitHub code repository for extension, debug and test.
- Develop specialized EDC Connector bridges, if necessary, to support local computing and system software environments. As an example, a "technology extension" bridge may be required to integrate a .NET application with the java-based EDC Connector.
- Debug and test the extended EDC Connectors accessing the Dataspace components, e.g. other Connectors, managed by the Dataspace Authority
- Release tested and verified extended EDC Connectors and Federated Catalog to the sandbox GitHub code repository for distribution to Companies / Organizations that need to be participants in the Rail Dataspace and wish to run them at their local installations.

6.6.4. Rail Dataspace Testing using the Sandbox Environment

With extended EDC Connectors, Federated Catalog and auxiliary mock services available in the Sandbox Environment end-to-end tests cycles are executed to validate the functionality of the Dataspace:

Publishing data assets:

- Contract Definitions are created by the data owner Participant for all parts (individual data sources) of the Data Asset, such as:
    - "Can be accessed only by a given member company's partners (Access Policy)
    - "Must be stored in Europe and used only for maintenance purposes (Usage Policy")
- The data owner Participant creates Data Asset Entry metadata description in the Federated Catalog
    - The Data Asset Entry is not the actual data source: it is a pointer to where the Data Asset is stored (locally to the Participant)
    - The Federated Catalog automatically "associates" the Data Asset with the Contract Definition
- The Data Contract (Data Asset + Contract Definition) is now available to other Participants that satisfy the policies contained.

Discovering data assets

- Scheduled Federated Catalog Crawler (FCC) uses the Dataspace members Registry to find other Participants
- FCC starts crawling the list of Participants
- for each Participant FCC reads registry entry which provides all information necessary to access the services of other Participants
- FCC proves its dataspace membership and Access Policy attributes to other Participants through the issued Verifiable Credential and the self-description of the Participant
- The FCC crawls the Federated Catalog from other Participants for available (published) Data Contracts
- If the credentials and self-description fulfil the Access Policy of a Data Contract, it becomes visible to the requesting Participant in their FCN
- The FCC caches the results of periodic crawling in the local Federated Catalog and provides means to query the local cache.
- Participants query their local Federated Catalog for Data Contracts available in the dataspace

Data assets exchange

- A Participant queries their Federated Catalog cache for available Data Contracts
- The Participant selects which Data Contract Offers it needs to consume
- The providing and the consuming Participant negotiate the Data Contract Agreement
- The contract negotiation may be automatic or involve manual workflow
- When the contract negotiation is completed, a Contract Agreement is created for the requested Data Contract and it's preserved for future audit. The Data Contract Agreement contains the Data Contract which contains the Usage Policy.
- The data consumer Participant initiates a data transfer request with the EDC Connector from the providing Participant
- The Connector component orchestrates the data transfer using specific data transfer technologies, depending on the extensions in use and the underlying data storage and processing technologies.
- The consuming and providing Connectors jointly orchestrate the transfer of data associated with the Data Asset
- Both Connectors record an audit history of the transaction

6.6.5. Onboarding of a Participant in the Dataspace

- The Participant applies with the Dataspace Authority using the registration custom Web App. An entry is created in the Dataspace member registry run by the Dataspace Authority and a Verifiable Credential is issued to the participant, proving membership in the dataspace.
- The Participant chooses and deploys EDC components from the sandbox's GitHub code repository

  - EDC Connector – orchestrates the sharing and transfer of data securely with other participants. The participant configures the deployment with the received

Verified Credential. The EDC Connector may be deployed in the cloud, on premises, or in a hybrid model. It can also be deployed across multiple clouds (for companies using more than one cloud provider)

- Federated Data Catalog and Crawler- advertises data to other participants and discovers data made available by other participants. The Federated Catalog is configured and the crawler scheduled.

7. Federated data space “Sandbox-Environment” Procurement

The considerations that drive the creation of a Sandbox Environment using available Eclipse Dataspace components, namely the twin requirements of a) delivery in time to Flagship Projects and b) consistency ‘by design’ with the European Data Strategy through reuse of existing open-source software and specialist experts apply to its procurement.
Since the start of the MOTIONAL project Work Package 31 participants have been researching available options, taking into account ‘movements’ in the market and open-source community as the European Dataspaces strategy increases in momentum and large Vendors develop specific offerings.

The decision has been made by WP 31 participants to source an ‘out of the box’, ready-to-run, ready-to-use, delivered as-a-service installation of the Eclipse Dataspace Components by T-Systems of Germany, which is a member of the Eclipse Dataspace Components development community and can, in addition, supply GAIA-X compatible Identity Provider services and issue Verifiable Credentials as required by the GAIA-X Trust Framework.

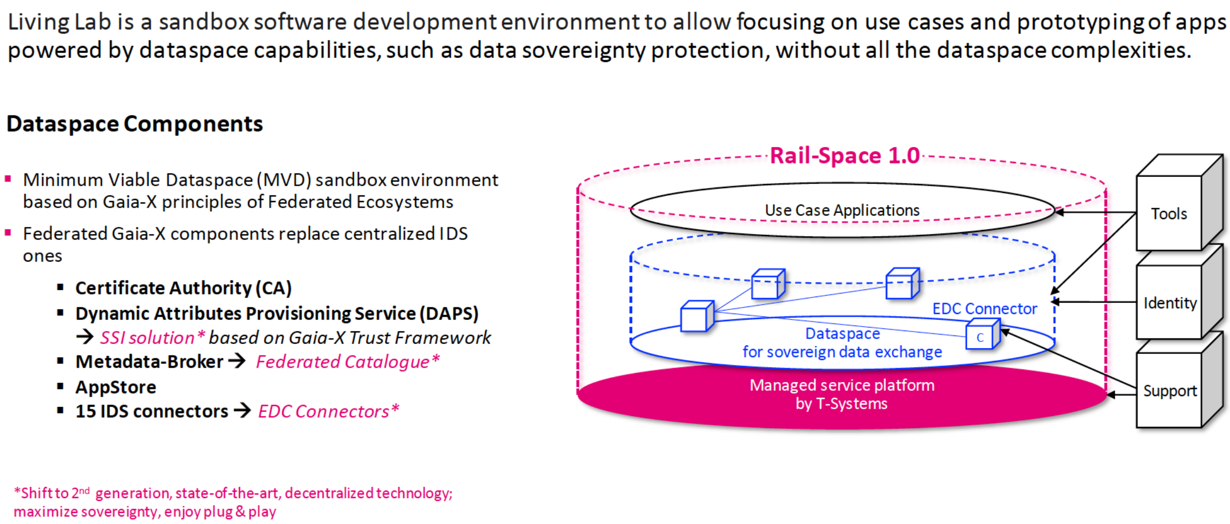

Figure 11 - "Living Lab" minimum viable dataspace

The Living Lab part of T-Systems “Data Intelligence Hub” commercial offering on which nine sectorial dataspaces, including GAIA-X Lighthouse Projects such as Catena-X, EONA-X and the Mobility Dataspace, are currently in development[23].

However, the development of the Rail Dataspace must include the creation of additional software artifacts, such as the auxiliary behavioural and computational ‘mock’ services, the necessary ‘bridges’ to company-specific computational environment described in section 6.6.3.

These additional artifacts are not a constituent part of the dataspace nor of the Eclipse dataspace components, they belong to MOTIONAL Consortium partners and must be available according to the provision of the project’s Grant Agreement. In addition, it cannot be assumed that the future

[23] Cfr: https://dih.telekom.com/en/

production Rail dataspace, not specified at this time, will still be hosted or operated by this particular Vendor. Rail dataspace specific products cannot therefore be 'released' into a commercial sandbox procured by Consortium partners through their own in-kind additional activity contribution to the project.

To meet these additional requirements while at the same time leveraging the availability of a read-to-use minimum viable dataspace the Sandbox Environment will therefore be complemented and extended by a complementary 'extension' environment provided by FS Technology hosting a GitHub code repository / version management installation, and the development and execution of the additional extension software artifacts. Services hosted in the extension sandbox environment behave as specialized Participants, such as semantic data transformation nodes, in the Living Lab Dataspace through EDC Connectors.

The extended Sandbox Environment is depicted in the figure below:

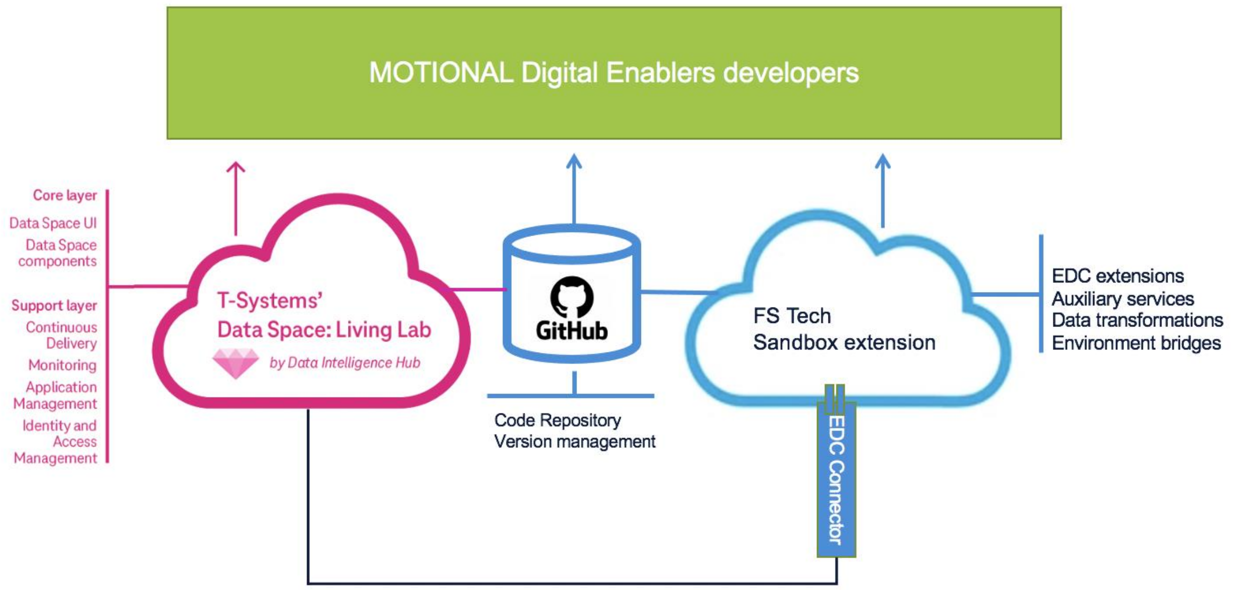

Figure 12 - Extended Sandbox Environment

8. Federated data space "Sandbox-Environment" Delivery

The delivery program for the extended Sandbox-Environment is depicted in the following figure:

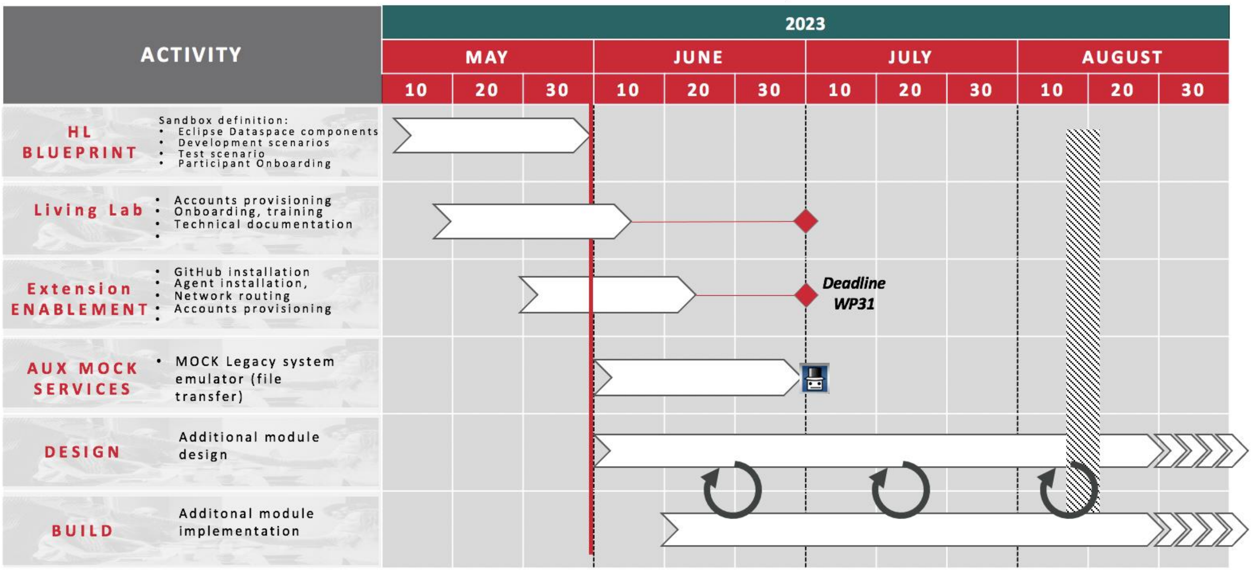

Figure 13 - Extended Sandbox delivery program

The High-level Blueprint describes the main decisions on what constitutes a Sandbox environment as a minimum viable dataspace used to build a Rail Dataspace to support data assets exchanges necessary for the realization of the Europe's Rail Joint Undertaking use cases and demonstrators. This High-level Blueprint is chapter 6 of this document, describing:

- The choice of Eclipse Dataspace components as the underlying implementation stack for the Dataspace concept described in section 5.
- The process for establishing the dataspace using Eclipse Dataspace components, including
    - Development scenarios on the sandbox
    - Testing scenarios on the sandbox
    - Onboarding of Participants on the dataspace

The Living Lab activity is the activity of making the Living Lab available and instructing developers on their use. Since the Living Lab part of the extended Sandbox Environment is a ready-to-use out-of-the-box installation its delivery consists essentially in the provision of accounts to the MOTIONAL developers. Onboarding instructions and technical documentation is also expected to be made available by the provider for establishing the connections with the FS Technology provided extension environment.

The Extension Enablement activity consists in setting up and configuring the FS Technology provided extensions, namely the installation and configuration of the GitHub code repository and version management software, the provisioning of accounts and configuring the network to allow secure and reliable access by MOTIONAL developer.

An initial design and implementation of an auxiliary mock service on the extension sandbox is also part of the extended sandbox deliver, providing a means to exercise the Living Lab with an actual end-to-end data exchange process from a remote connector, thus verifying the completeness and readiness status of the extended sandbox environment .

9. Appendix – Example data exchange scenario through EDC Connectors

In this appendix a data exchange scenario is presented realized in actual software based on the standard open-source EDC Connector.
The purpose of the exercise is the following:

- Demonstrate how 'legacy' systems, i.e. business applications, can interact with the EDC Connector through REST web services exposed by the Connector itself. While it is entirely possible to integrate the connector within the business applications, leveraging the exposed web services is the recommended approach in that it provides loose coupling between independent executables which may be based on different underlying system software (e.g. a .NET application communicating with the java-based Connector), supports usage of the same connector by different applications, and expands the available options for deployment and systems management.
- Show that the Consumer and Provider are actually independent entities: only the Connectors know about each other, but the Consumer is unaware of what system produces the data or where it is, and likewise the Producer is unaware of what system consumes it or where it is.
- Provide actual representative json messages of how asset, policy, contract definitions, contract negotiations and transfer request metadata are defined and exchanged by the Connector's control-plane
- Provide two different scenarios of the data exchange pattern through the Connector's data plane: data is pushed by the provider to the consumer, or data is pulled by the consumer from the provider.

The scenario is described by the following figure:

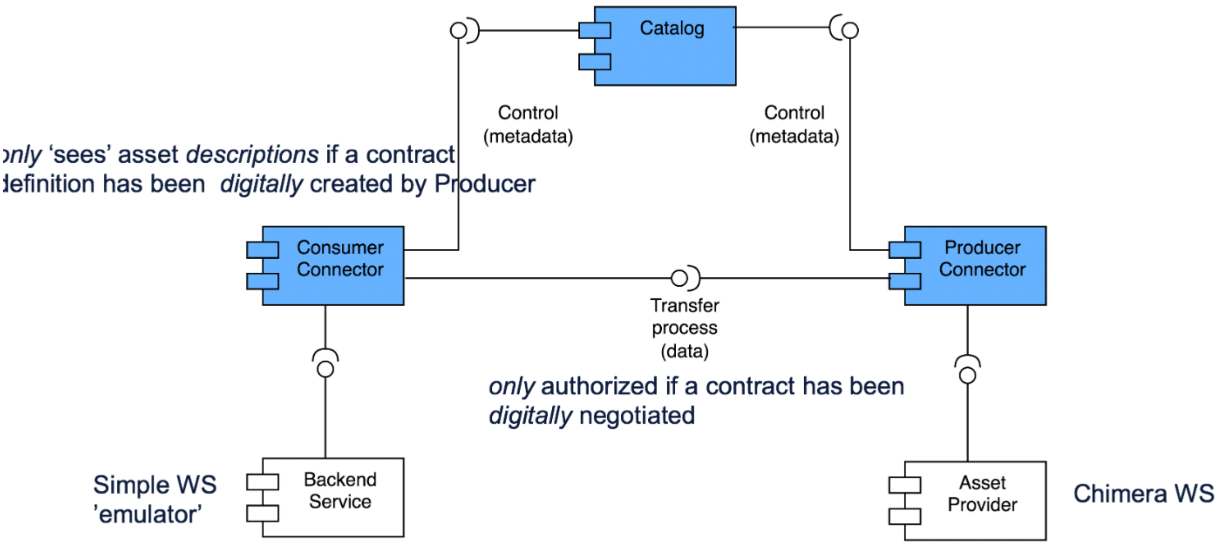

Figure 14 - Example dataspace exchange scenario

The elements in blue are EDC dataspace components.
The Asset Provider is an application that performs semantic conversion of a sample infrastructure topology file to the System Pillar – defined 'Rail Topology" common data model: it exposes a web service that the Producer Connector calls to retrieve the data for exchange. The metadata *description* of this data is published in the Catalog by the provider, who also defines a policy and contract definition for the provision of the actual data.

The Asset Provider's web service, and therefore the actual data, is invisible to the Consumer: the latter can only interact with the Consumer Connector which can only get the data from the Provider Connector. Only the Provider Connector has knowledge of the Asset Provider. This means that the Cosumer does not know where the actual data Is stored and in which manner is produced, and it means additionally that the Provider could move the data and/or the Asset Provider to some other environment, such as a cloud, without the Consumer being affected.

The Backend service emulates an Application that needs the data from the Provider. It calls Consumer Connector web services to query the Catalog to find the metadata description of the asset it needs. It then calls Consumer Connector web services to negotiate a contract with the Provider. When the negotiation is finalized, it retrieves the Provider-supplied contract id, and uses it to either request the provider to 'push' the actual data to the Backend Service, or to obtain an authorization token with which to 'pull' the data from the Provider.

9.1. Push Transfer

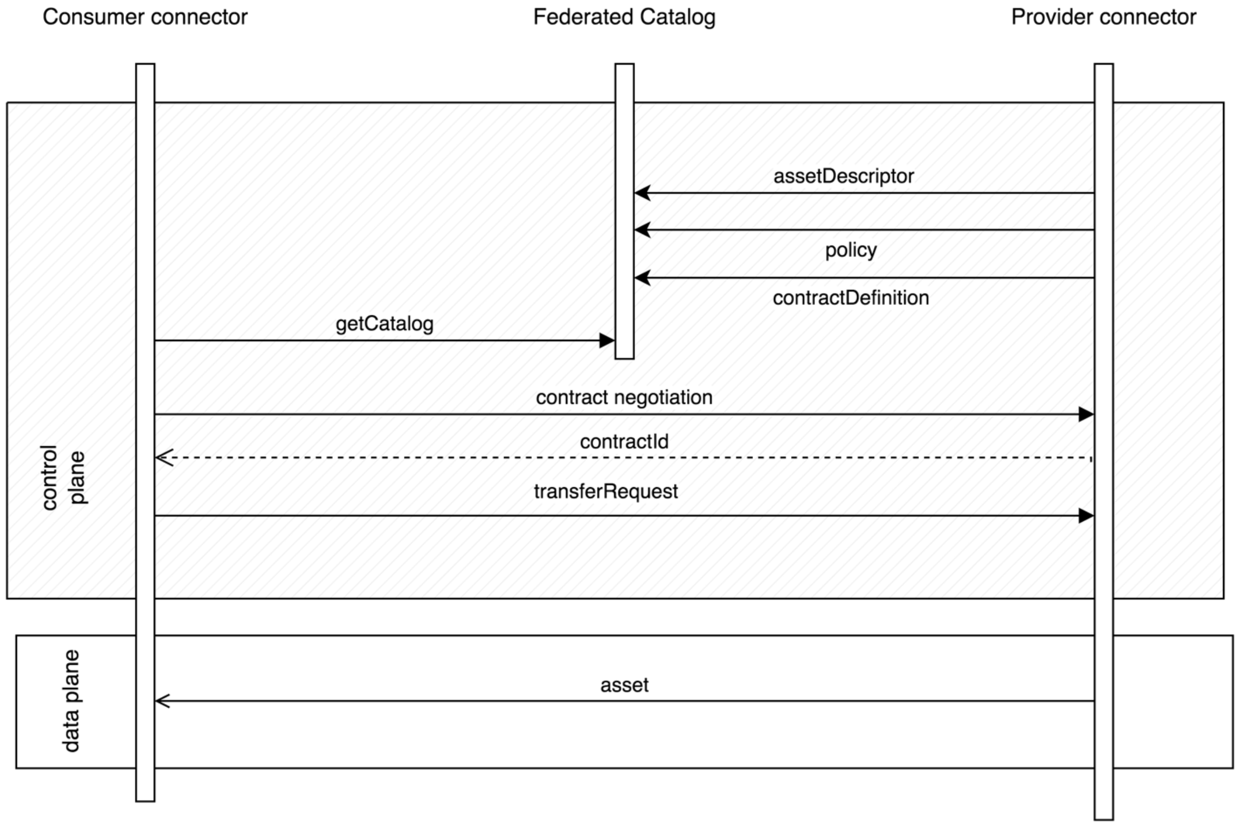

Figure 15 - Push exchange pattern

Publish Asset description metadata (request)


```
{
    "@context": {
        "edc": "https://w3id.org/edc/v0.0.1/ns/"
    },
    "asset": {
        "@id": "trackedgelink",
        "properties": {
            "name": "Baltic Network TrackEdgeLinks",
            "contenttype": "application/json",
            "model" : "SD1 - Topology Model"
        }
    },
    "dataAddress": {
        "properties": {
            "name": "Chimera semantic transformer",
            "baseUrl": "http://localhost:8081/rml/erju/trackedgelinks",
```

```
            "type": "HttpData"
        }
    }
}
```

Publish a Policy (request)


```
{
    "@context": {
        "edc": "https://w3id.org/edc/v0.0.1/ns/"
    },
    "@id": "aPolicy",
    "policy": {
        "@context": "http://www.w3.org/ns/odrl.jsonld",
        "@type": "set",
        "permission": [
            {
                "target": "http://localhost:8081/rml/erju/trackedgelinks",
                "action": "use"
            }
        ],
        "prohibition": [],
        "obligation": []
    }
}
```


Get Catalog (response)


```
{
    "@id": "d4e9a0ab-433b-4272-91bf-c5b14487716d",
    "@type": "dcat:Catalog",
    "dcat:dataset": {
        "@id": "8f07c710-d9d4-4680-9fe8-2c5329d1f118",
        "@type": "dcat:Dataset",
        "odrl:hasPolicy": {
            "@id": "1:trackedgelink:d41ff1b7-f443-44c6-9931-80a23ee00e80",
            "@type": "odrl:Set",
            "odrl:permission": {
                "odrl:target": "trackedgelink",
                "odrl:action": {
                    "odrl:type": "http://www.w3.org/ns/odrl/2/use"
                }
            },
            "odrl:prohibition": [],
            "odrl:obligation": [],
```

```
            "odrl:target": "trackedgelink"
        },
        "dcat:distribution": [
            {
                "@type": "dcat:Distribution",
                "dct:format": {
                    "@id": "HttpProxy"
                },
                "dcat:accessService": "242afea2-dc8b-4e2d-bf7d-33dad964f1ca"
            },
            {
                "@type": "dcat:Distribution",
                "dct:format": {
                    "@id": "HttpData"
                },
                "dcat:accessService": "242afea2-dc8b-4e2d-bf7d-33dad964f1ca"
            }
        ],
        "edc:name": "Baltic Network TrackEdgeLinks",
        "edc:model": "SD1 - Topology Model",
        "edc:id": "trackedgelink",
        "edc:contenttype": "application/json"
    },
    "dcat:service": {
        "@id": "242afea2-dc8b-4e2d-bf7d-33dad964f1ca",
        "@type": "dcat:DataService",
        "dct:terms": "connector",
        "dct:endpointUrl": "http://localhost:19194/protocol"
    },
    "edc:participantId": "provider",
    "@context": {
        "dct": "https://purl.org/dc/terms/",
        "edc": "https://w3id.org/edc/v0.0.1/ns/",
        "dcat": "https://www.w3.org/ns/dcat/",
        "odrl": "http://www.w3.org/ns/odrl/2/",
        "dspace": "https://w3id.org/dspace/v0.8/"
    }
}
```

Negotiate Contract (request)

```
{
    "@context": {
        "edc": "https://w3id.org/edc/v0.0.1/ns/"
    },
    "@type": "NegotiationInitiateRequestDto",
    "connectorId": "provider",
    "connectorAddress": "http://localhost:19194/protocol",
    "consumerId": "consumer",
```

```
    "providerId": "provider",
    "protocol": "dataspace-protocol-http",
    "offer": {
        "offerId": "1:trackedgelink:d41ff1b7-f443-44c6-9931-80a23ee00e80",
        "assetId": "trackedgelink",
        "policy": {
            "@context": "http://www.w3.org/ns/odrl.jsonld",
            "@id": "1:trackedgelink:2699c3c7-8291-44b9-b575-69b8ed2b13e8",
            "@type": "Set",
            "odrl:permission": {
                "odrl:target": "trackedgelink",
                "odrl:action": {
                    "odrl:type": "http://www.w3.org/ns/odrl/2/use"
                }
            },
            "prohibition": [],
            "obligation": [],
            "target": "trackedgelink"
        }
    }
}
```

Get Contract Id (response)

```
{
    "@type": "edc:ContractNegotiationDto",
    "@id": "ee0a326e-ac5e-4aa9-97a5-e8ac7fe80653",
    "edc:type": "CONSUMER",
    "edc:protocol": "dataspace-protocol-http",
    "edc:state": "FINALIZED",
    "edc:counterPartyAddress": "http://localhost:19194/protocol",
    "edc:callbackAddresses": [],
    "edc:contractAgreementId": "1:trackedgelink:ba41edb7-b11d-48a6-a53a-d2a5e55b3b09",
    "@context": {
        "dct": "https://purl.org/dc/terms/",
        "edc": "https://w3id.org/edc/v0.0.1/ns/",
        "dcat": "https://www.w3.org/ns/dcat/",
        "odrl": "http://www.w3.org/ns/odrl/2/",
        "dspace": "https://w3id.org/dspace/v0.8/"
    }
}
```

Initiate push Data Transfer (request)

```
{
    "@context": {
        "edc": "https://w3id.org/edc/v0.0.1/ns/"
```

```
    },
    "@type": "TransferRequestDto",
    "connectorId": "provider",
    "connectorAddress": "http://localhost:19194/protocol",
    "contractId": "1:trackedgelink:ba41edb7-b11d-48a6-a53a-d2a5e55b3b09",
    "assetId": "trackedgelink",
    "managedResources": false,
    "protocol": "dataspace-protocol-http",
    "dataDestination": {
        "@type": "DataAddress",
        "type": "HttpData",
        "properties": {
            "baseUrl": "http://localhost:4000/api/consumer/store"
        }
    }
}
```

The baseUrl property of the dataDestination is the web service that will be called by the Producer Connector to ‘push’ the data to.

Data pushed to Backend System (extract)

```
{
  "$schema" : "ERJU meta-model.json",
  "isDefinedBy" : "http://ERJU/datamodel/0.1/railtopo",
  "name" : "railtopo",
  "prefix" : "topo",
  "intId" : 1,
  "version" : "0.1",
  "info" : "Railway network topology and topography, position wrt topology, location on a map, geometry, aka. alignment)",
  "structs" : [

                {
                  "name" : "TrackEdgeLink",
                  "info" : "Defines a relation between two track edges along which a train can run.",
               "see" : "http://ontorail.org/rsm12/Common/Topology/PositionedRelation",
          "belongsToSubPackage" : "topology",

           "attrs" : [
      {
        "intId" : 1,
        "name" : "<http://ERJU/datamodel/0.1/railtopo/TEL00070100-0103-010d-000e-06010f02040800070100-0103-040d-000e-06010f02050c>",
        "dataType" : "IRI",
        "key" : true,
        "info" : "Identity for referencing, e.g. by points and crossings"
      },
      {
        "intId" : 2,
        "name" : "trackEdgeA",
        "reference" : "<http://ERJU/datamodel/0.1/railtopo/link/L00070100-0103-010d-000e-06010f020408>",
        "info" : "Connects to track edge A"
      },
      {
        "intId" : 3,
        "name" : "trackEdgeB",
        "reference" : "<http://ERJU/datamodel/0.1/railtopo/link/L00070100-0103-040d-000e-06010f02050c>",
        "info" : "Connects to track edge B"
      }
    ]
                }  ,
```

•••••

```
                {
                  "name" : "TrackEdgeLink",
                  "info" : "Defines a relation between two track edges along which a train can run.",
               "see" : "http://ontorail.org/rsm12/Common/Topology/PositionedRelation",
          "belongsToSubPackage" : "topology",

           "attrs" : [
        {
          "intId" : 1,
          "name" : "<http://ERJU/datamodel/0.1/railtopo/TEL0f060100-0e06-0508-000e-050b0d070b040f060100-0509-0209-000e-050b0d070c08>",
          "dataType" : "IRI",
          "key" : true,
          "info" : "Identity for referencing, e.g. by points and crossings"
        },
        {
          "intId" : 2,
          "name" : "trackEdgeA",
          "reference" : "<http://ERJU/datamodel/0.1/railtopo/link/L0f060100-0e06-0508-000e-050b0d070b04>",
          "info" : "Connects to track edge A"
        },
        {
          "intId" : 3,
          "name" : "trackEdgeB",
          "reference" : "<http://ERJU/datamodel/0.1/railtopo/link/L0f060100-0509-0209-000e-050b0d070c08>",
          "info" : "Connects to track edge B"
        }
      ]
                }
  ]

}
```

9.2. Pull Transfer

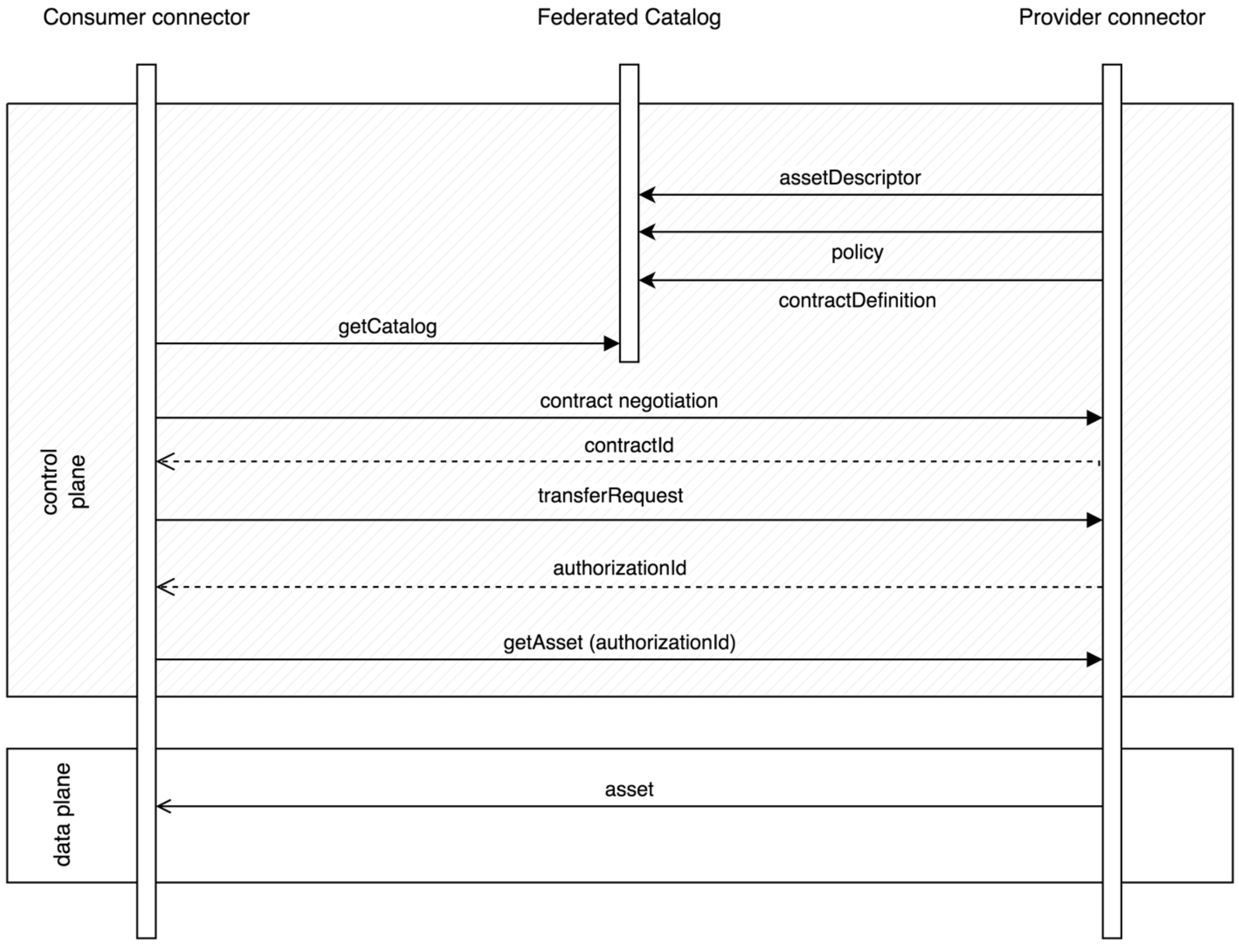

Figure 16 - Pull exchange pattern

In the pull transfer scenario the publication of assets description and policy metadata, the query on the catalog, the contract negotiation and the request for the contract id are the same as in the push scenario. In the pull scenario, however, the Backend Services does not receive the data directly to a data destination the Consumer Connectot specifies in the Transfer request: instead it receives an authorizazion code which the Consumer Connector can use to retrieve the data.

Initiate pull Data Transfer (request)

```
{
    "@context": {
        "edc": "https://w3id.org/edc/v0.0.1/ns/"
    },
    "@type": "TransferRequestDto",
    "connectorId": "provider",
    "connectorAddress": "http://localhost:19194/protocol",
```

```
    "contractId": "1:trackedgelink:ba41edb7-b11d-48a6-a53a-d2a5e55b3b09",
    "assetId": "trackedgelink",
    "managedResources": false,
    "protocol": "dataspace-protocol-http",
    "dataDestination": {
        "@type": "DataAddress",
        "type": "HttpProxy"
    }
}
```

Since this is a pull transfer request the dataDestination property does not specify a web service to call to push the data.

Authorization token (received)

```
{
    "id": "face4d4f-e739-4ee0-847d-a4d251eda86e",
    "endpoint": "http://localhost:29291/public/",
    "authKey": "Authorization",
    "authCode": "eyJhbGciOiJSUzI1NiJ9.eyJleHAiOjE2ODczNjA5OTksImRhZCI6IntcInByb3BlcnRpZXNcIjp7XCJhdXRoS2V5X(
    "properties": {
        "https://w3id.org/edc/v0.0.1/ns/cid": "1:trackedgelink:ba41edb7-b11d-48a6-a53a-d2a5e55b3b09"
    }
}
```

On sending the authorization code in the Authorization header of the call to the Provider Connectors’ web service the Consumer receives the same data as in the push scenario

10. Bibliography

OPEN DEI: “Design Principles for Data Spaces – Position Paper” https://design-principles-for-data-spaces.org/

IDSA: “IDS Reference Architecture Model version 4.0 https://docs.internationaldataspaces.org/ids-ram-4/

A. Braud, G. Fromentoux, B. Radier and O. Le Grand, "The Road to European Digital Sovereignty with Gaia-X and IDSA," in *IEEE Network*, vol. 35, no. 2, pp. 4-5, March/April 2021, doi: 10.1109/MNET.2021.9387709.

Nagel, Lars, and Douwe Lycklama. "Design principles for data spaces." In *Position Paper*. International Data Spaces Association, 2021

Otto, Boris. "A federated infrastructure for European data spaces." *Communications of the ACM* 65, no. 4 (2022): 44-45.

Kraemer, Peter, Crispin Niebel, and Abel Reiberg. "Gaia-X and Business Models." (2023).

Kremer, Marco, Lucas Pohling, Henning Gösling, Christoph Heinbach, Timon Sachweh, Sonika Gogineni, and Kolja Berger. "An Intelligent Arrival Time Prediction Service in a Federated Data Ecosystem: the Minimum Viable Demonstrator of the GAIA-X 4 ROMS Research Project." *Available at SSRN 4331859* (2022).

Scerri, S., T. Tuikka, and I. Lopez de Vallejoan. "Towards a European data sharing space." *Report. Big Data Value Association* (2020).

Tardieu, Hubert. "Role of Gaia-X in the European Data Space Ecosystem." In *Designing Data Spaces: The Ecosystem Approach to Competitive Advantage*, pp. 41-59. Cham: Springer International Publishing, 2022.

Guo J, Cheng Y, Wang D, Tao F and Pickl S. (2021). Industrial Dataspace for smart manufacturing: connotation, key technologies, and framework. International Journal of Production Research. 10.1080/00207543.2021.1955996. 61:12. (3868-3883). Online publication date: 18-Jun-2023.

Nakrani, Pintukumar, M. Sc Jonas Urs Schlenger, and Christof Duvenbeck. "Conceptual Framework of Construction Data Storage using Gaia-X Federation Services: Demonstration with Usecase of Project iECO." (2022).

Europe’s Quest for Digital Sovereignty: GAIA-X as a Case Study Author(s): Simona Autolitano and Agnieszka Pawlowska Istituto Affari Internazionali (IAI) (2021)

Hoffmann, F.; Weber, M.; Weigold, M.; Metternich, J.: Developing GAIA-X Business Models for Production. In: Herberger, D.; Hübner, M. (Eds.): Proceedings of the Conference on Production Systems and Logistics: CPSL 2022. Hannover : publish-Ing., 2022, S. 583-594. DOI: