TNO innovation for life

Douane Pilot 2024

FIT Gap Analysis

TNO Publiek › TNO 2024 R12233
20 november 2024

ICT, Strategy & Policy
www.tno.nl

TNO 2024 R12233 – 20 november 2024

Douane Pilot 2024

FIT Gap Analysis

Auteurs	T. Chirvasuta, T. Nijman, H. van Nieuwenhuijze
Rubricering rapport	TNO Publiek
Titel	TNO Publiek
Rapporttekst	TNO Publiek
Aantal pagina's	25 (excl. voor- en achterblad)
Aantal bijlagen	0
Projectnaam	Pilot Douane Singapore
Projectnummer	060.61355

Alle rechten voorbehouden
Niets uit deze uitgave mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm of op welke andere wijze dan ook zonder voorafgaande schriftelijke toestemming van TNO.

© 2024 TNO

Table of Contents

1 Executive Summary

Dutch customs currently uses the FEDeRATED prototype node for data sharing solution based on the FEDeRATED vision [1]. This FEDeRATED prototype is based on CORDA technology [2], which works well but has some drawbacks. The FEDeRATED node uses CORDA 4.9.2 and upgrading to CORDA 5 will require significant effort. Moreover, CORDA is not an open standard and in order to prevent lock in to a proprietary solution, it is important to consider alternatives that are future-proof (based on open standards), and aligns with other initiatives within the Netherlands and Europe. Since Europe is moving rapidly towards open standards in the area of Data Spaces, this report compares the current implementation in CORDA 4.9.2 with the Eclipse Data Space Components (EDC) [4]. Furthermore, it compares such a solution with the 'basic data infrastructure' (BDI [3]).

The advice is to replace the current solution, based on CORDA, with a solution based on the Eclipse Dataspace Components.

This fits well with the desire of Dutch Customs and the Ministry of Infrastructure and Water management to align with international standards and national initiatives. Using a more commonly used basis for the software will allow Customs to more easily manage and use the prototype. The consequences of choosing an EDC-based solution, which will be run and managed by Customs, are:

- Advantages of an EDC-based solution
  - Knowledge about EDC and the underlying programming language is more widespread, allowing for easier management and maintainability.
  - Data spaces are becoming a preferred structure for data sharing in Europe.
  - This approach aligns better with other initiatives in logistics.
  - Open source software is preferred by the Dutch government.
- Impact for the Customs organization
  - The software will run in an Azure environment [5]. This must be available for Customs and sufficient knowledge to manage such an environment must be present.
  - Onboarding of new participants in the solution requires configuration of message translations. This can be done efficiently using the Semantic Treehouse software, which will require knowledge about the process and techniques used [6].
  - New participants require a confirmed identity. In CORDA identification and access was provided automatically and implicitly, in the new solution this requires (some) human interaction
  - The EDC is open source software under active development. This means that adopting new versions may require more work than with a commercially supported infrastructure.

- Additional features available in CORDA. These features are neither used nor required in the current solution for Customs.
    - In Corda advanced rules for automatic routing of events may be specified
    - Corda provides a strong system for non-repudiation, providing proof of data exchange

Replacing the current solution with a solution based on the EDC could be achieved in a controlled manner by:

- Deploying the current solution in an environment managed by Customs, allowing the organisation to build up some experience and decreasing reliance on the (testing) environment provided by TNO.
- Deploying the newly developed solution in a separate environment once it is available, to allow for testing and uninterrupted availability
    - It could even be possible to run both solutions in parallel, depending on the integration with the other participants
- Connecting all participants to the new solution, depending on the choices in deployment this could be transparent for all participants.

With the previous steps the current, CORDA-based, solution would remain available as a backup solution while the new solution is deployed. Once the new solution has proved itself the backup can be mothballed.

2 Summary

The evolving landscape of data sharing in the European Union, driven by legislative initiatives such as the EU Data Act and Data Governance Act, has prompted Dutch Customs to explore more standardized and interoperable solutions for data sharing. Currently, Dutch Customs operates a FEDeRATED prototype built on Corda Distributed Ledger Technology (DLT), designed to enhance efficiency in executing legal responsibilities (FEDeRATED-CORDA). However, emerging generic data space technologies offer a promising alternative (FEDeRATED-EDC), aligning with international standards and fostering greater flexibility, scalability, and interoperability.
This fit-gap analysis compares FEDeRATED-CORDA with a potential architecture built on generic data space technology, FEDeRATED-EDC. The goal is to assess the differences in functionality between these two approaches. Additionally, a high level compareison was made between the 'basic data infrastructure' (BDI [6]) and FEDeRATED-EDC, to assess interoperability with BDI.
Key findings include:

- Corda's Strengths:
  - Corda excels in advanced data routing and seamless event distribution.
  - It offers a built-in mechanism for pushing events to designated recipients.
  - Supports selective data sharing and non-repudiation by default through cryptographic hashes and a notary node
- Generic Data Space advantages:
  - Adheres to international standards like the Data Space Protocol [7] and Decentralized Claims Protocol [8].
  - Provides greater flexibility, enhanced interoperability, and superior data sovereignty capabilities.
  - Supports dynamic onboarding and scalability without vendor lock-in.
- Gaps identified
  - Corda features advanced rules for automatic data distribution and routing, generic data space components do not facilitate this feature out of the box.
  - While Corda facilitates data exchange directly, generic data spaces only facilitate the exchange, requiring additional components to transfer the data.

The goal of this analysis is to help Dutch Customs determine whether transitioning to a standardized, generic data space infrastructure would provide long-term benefits in terms of interoperability and adaptability, outweighing some added complexity and overhead.

3 Introduction

Dutch Customs has developed a prototype data-sharing solution (FEDeRATED-CORDA) to evaluate which types of data can be provided by organisations in the logistics domain to support the efficient and reliable execution of Customs legal responsibilities. The current proof-of-concept system for data sharing in use by Customs was developed in line with the design principles of the Digital Transport and Logistics Forum (DTLF)[9] as expressed in the FEDeRATED project [1]. The data is structured according to the FEDeRATED semantic model for event-based data exchange [10] and primarily focuses on ship movements.
This implementation uses Corda, a type of Distributed Ledger Technology (DLT), as the data exchange platform[2]. This implementation has several key problems. It is built on version 4 of Corda, an older version approaching end-of-life, meaning it will soon no longer receive updates or security patches. While an upgrade to version 5 is possible, this would require significant effort. Moreover, Corda is a closed-source platform and does not support the latest data-sharing standards being developed through various international standardization efforts. Since Corda is proprietary, vendor lock-in is also a real concern. This prototype is referred to as FEDeRATED-CORDA.
Therefore, Dutch Customs has expressed a desire to transition towards an architecture that aligns with international state-of-the-art data sharing standards and components—referred to as ”generic data sharing components” and would like to know if this is a worthwhile effort to upgrading to Corda 5, solving the problems mentioned.
Continuing the use of semantic data in the data-sharing solution through user friendly tooling is a requirement. In the current solution the vocabulary hub solution ‘Semantic Treehouse’ [6] has been used for ontology management and data format translation templates. Continued use of this solution is desirable.

The landscape of data sharing has evolved rapidly in recent years, driven by the increasing demand for accessible and interoperable data. In the European Union, innovation in this area has been growing steadily, largely spurred by a top-down approach where legislative initiatives aim to foster progress in data sharing. Key examples include the EU Data Act [11], the Data Governance Act [12], and the broader EU Strategy for Data [13]. These frameworks have led to the emergence of several European initiatives, both public and private, such as the Data Spaces Support Centre [14] and the International Data Spaces Association [15].
Given the large-scale support for these initiatives, aligning with them is advantageous.

The goal of this fit-gap analysis is to evaluate how an implementation based on generic data space standards and components would differ from the current system. This analysis identifies how such an implementation would fit the current implementation, what the missing features are and, where possible, suggest alternative ways to realize missing features.

This analysis will enable Dutch Customs to make an informed decision on whether it is worthwhile to re-engineer the current system using generic data space components and standards. This is referred to as FEDeRATED-EDC.

4 Design Principles

The FEDeRATED-CORDA prototype was designed based on the design principles as stated by the DTLF [6]:

- Plug and play – the solution proposed has to be able to integrate in an inexpensive manner with existing systems;
- Technology independent services – the usability of the digital business service offering of logistics companies should not be bound to a technology implementation to enable interoperability;
- Federation – ensuring the sovereignty of the data shared by the logistic partners is crucial when considering the adoption of the Customs solution suite;
- Safe, secure and trusted - creating trust among all stakeholders is crucial for the onboarding of new collaborators to the network.

In addition to the design principles, the requirements from Customs were:

- Data at source, events can be sent to each participant and contain very limited data. An API is provided by each node allowing the retrieval of additional data;
- Nodes can actively provide data to backend systems of participants;
- Events and message models can be modelled using a shared ontology;
- An API is available to find available data;
- A semantic adapter is provided allowing backend systems to call an API using common syntax (JSON) and providing a mapping to the shared language;
- Event routing is configurable;
- API's are secured using HTTP Basic Authentication.

5 FEDeRATED Masterplan

The design principles of the DTLF are captured in the FEDeRATED masterplan, which was the vision towards which the current implementation was a first step [1]. FEDeRATED aims to resolve the challenges of real time data exchange between supply chain stakeholders and with authorities. Functional requirements and the implementation components researched during the FEDeRATED project are described.

6.1 Functional Requirements

The DTLF design principles have motivated the need of a data sharing network in which each participant of the network may join in an inexpensive manner, is able to use the technology of their choice, and securely share and receive data from network collaborators. The separation of the connection to the data sharing network from the connection with the participants' backend system has motivated the research in the FEDeRATED project of a cloud-based gateway software solution. This gateway handles the registration and subsequent authentication of the user in the data sharing network; the gateway provides visibility of the registered users to one another ; the gateway interfaces with the backend system of the end-user; and it distributes data to be shared in the network. We detail the 4 functionalities of the gateway:

- The registration of the end-user to the network is automatically handled when deploying an instance of the gateway. Upon subsequent participation in the network, the gateway handles the authentication of the end-user
- each participant of the data sharing network may see the contact details (name and IP address) of the gateway of other participant users, thus creating visibility between the data sharing network participants
- the interface from the data sharing network to the participants backend system (semantics wise) is a 2-step process, translating the data format of the participant by relating this to a shared semantic model. The process is automated through the use of data format translation templates exported from Semantic Treehouse and the open source RML Mapper [16] solution.
  The semantics interfacing process is detailed below
  - Step 1: translation of the message (an instance of data to be shared on the network) of the end-user, expressed with their proprietary vocabulary, to the network standardized vocabulary (JSON juggling)
  - Step 2: translation of the distribution[17] (JSON serialized) of the end-user message to the shared semantic model as 'triples' (RML mapping)
  - Distributing the data is handled by the gateway using the underlying the peer to peer technology, and is configured by the end-user in the gateway or overruled when sharing data.

6.2 FEDeRATED Masterplan Implementation

As mentiond, the FEDeRATED masterplan described a vision for data sharing in the domain of logistics, and as such it is very broad. The current implementation is far from complete, and has implemented the following elements:

- A large part of the FEDeRATED ontology, the shared semantic model, was modelled in Semantic Treehouse;
- Translation scripts from the end-user vocabulary to the standardized vocabulary in the network (FEDeRATED data model):
    - This can be performed locally, on the end-user system. An example was written in Python for translation of the NTP-proprietary vocabulary to the FEDeRATED vocabulary
    - An alternative implementation uses the cloud environment of the end-user. An example was written as an Azure script for the translation of the Portbase proprietary vocabulary to the FEDeRATED vocabulary
- Corda node implementation
    - participant registration - deploying a FEDeRATED node on the participant environment
    - participant authentication - user name and password authentication when accessing the deploying FEDeRATED node in the participant environment
    - participant gateway contact details - when a new FEDeRATED node is deployed, its contact details are saved on the ledger and may be retrieved by other participants from their own gateway
    - interface from the participants system to the network - distribution translation and API's for data sharing network participation
        - Distribution translation - data expressed in JSON is transformed to triples based on an RML data translation template, provided by the end-user. Such a specification can be extracted from Semantic Treehouse or written by hand.
        - OpenAPI specification - discovering network peers, sending data and requesting data (using SPARQL queries [18]) from other network peers
    - An interface from the network to the participant - Webhooks can be registered to provide updates to the backend system of the participant whenever new data is shared with them in the network
    - A visualization tool for browsing through the data shared in the network by the participant and to the participant

6 Current implementation

This section outlines the architecture of the current system, which is built on the Corda platform [2], a decentralized ledger technology (DLT) designed for secure and efficient data sharing.
The system utilizes Corda to manage distribution rules for network transactions through its flow mechanism. When a node pushes data to the ledger, the Corda flows process the data as part of the network transaction and distribute it according to predefined rules specified in the flow or according to the distribution specified explicitly by the originating node. Unlike traditional DLTs, which require that all information on the ledger be shared with the entire network, Corda allows nodes to selectively share events with specific counterparties. This selective data sharing means that some information can be distributed to different nodes at different times, depending on the rules established in the Corda flows. For example, in the context of the FEDeRATED vision, certain critical events—such as accidents—could trigger conditional data sharing among specific participants.

While Corda's selective sharing model offers flexibility, it also introduces challenges. Since not all nodes are aware of all transactions, this can undermine non-repudiation and make it difficult to verify events across the entire network. To address this, Corda introduces a key component: the 'Notary'. The Notary acts as a centralized ledger and ensures the integrity of transactions by requiring all data pushed to the ledger to be accompanied by a cryptographic hash shared with the notary.

A second key component is the 'Doorman' which manages network membership by tracking the nodes and overseeing the onboarding of new participants (providing the desired 'Plug and play' functionality). Finally the 'Network Map service' allows nodes to discover each other. Figure 6.1 illustrates the current architecture, highlighting the interactions between nodes (the doorman service is omitted as it plays no significant role after a node has been onboarded to the network).

Each node in the network exposes an API that participants can use to send events they wish to share. These events, represented as linked-data messages, are stored in a triple store, allowing for efficient retrieval. The API then forwards the messages to Corda (Distribution), where they are distributed according to the predefined flow logic.

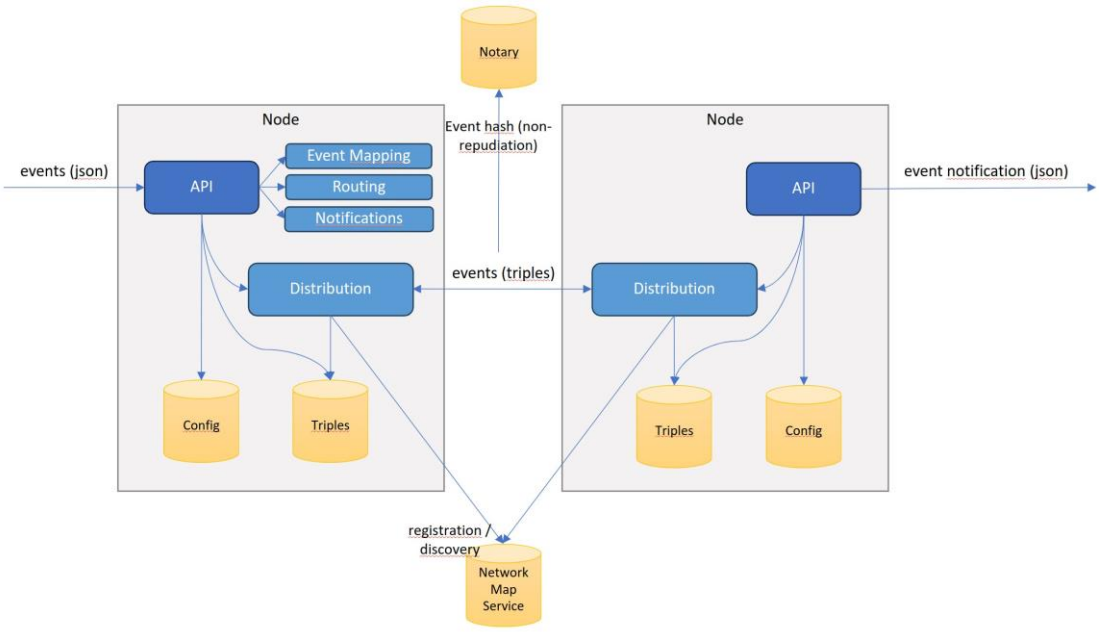

Figure 6.1 Current Pilot architecture

7 Current Requirements

This section deals with the requirements for the current system, as specified by Customs, defined in the FEDeRATED masterplan or implied by the DTLF principles. Not all requirements from the FEDeRATED masterplan have been considered for the current system. In the table below we present these requirements, whether it is functional or non-functional, the building block of DTLF it tackles and the description.

Table 7.1: System Requirements

ID	Type	DTLF	Description
FR-001	Functional	TIS	The system must support the exchange of event data. The system shall exchange ('push') data to recipients based on routing rules or explicit addressing. The exchange must occur within a predefined time of the event trigger ('(near) real time') and events contain limited data ('data at source').
FR-002	Functional	Plug and play	The system must allow new participants to complete a well-defined registration process. The process should be intuitive, and include clear instructions for each step. The system must validate all required information, such as personal details and contact information, before final submission.
FR-003	Functional	Trusted	The system must provide a trust framework that enables participants to establish secure connections for exchanging sensitive events.
FR-004	Functional	Federation	The system must allow nodes to automatically discover each other and discover available data within a reasonable amount of time after joining the network. In the event of a discovery failure, the system must attempt re-discovery within a predefined amount of time and log any errors for administrative review.
FR-005	Functional	Plug and play	The system must facilitate interfacing with backend systems to the data sharing system and provide semantic translation facilities for these interfaces.
NFR-001	Non-functional	n/a	The system must be scalable, allowing the addition of new nodes without causing network congestion. Ensuring that the system continues to perform efficiently as it scales.
NFR-002	Non-functional	Safe, secure	Data exchange must happen securely. The system must use industry-standard encryption protocols (such as TLS 1.3). API's must be secured with at least basic authentication.

8 Generic Data Space Technology

8.1 Generic data space components

The purpose of this section is to provide an introduction to ”generic data space components”. It addresses the current state-of-the-art in data sharing technology and how it can be used to create federated, interoperable data spaces.

”Generic Data Space components” refers to any software component that can be deployed within a data space, aligning with the Digital Europe Programme for common European Data Spaces and the European strategy for data. The Data Spaces Support Centre (DSSC), funded by the Digital Europe Programme, serves as the reference point [14]. The DSSC is responsible for the blueprint for Common European Data Spaces [19]. It provides a comprehensive overview of the technical and organizational building blocks.

The DSSC defines a data space as follows: ”A data space is a distributed system defined by a governance framework that enables trustworthy data transactions between participants while supporting trust and data sovereignty. A data space is implemented by one or more infrastructures and supports one or more use cases”

A data space does not refer to a fixed architecture. Rather, it refers to an entire family of architectures that enable sovereign data sharing. It is therefore impossible to include every design choice here. Nevertheless, a general description can still be provided. The terminology used here is based on the conceptual model for data spaces by the DSSC [20].

A data space starts with a governance framework. This establishes at least the rules of the data space. It may also include other things such as:

- The functionalities of the data space
- Processes of the data space
- Roles defined in the data space

The Data Space Authority, a role fulfilled by one or more participants, is responsible for maintaining, developing and enforcing the data space governance framework. Some participants may provide other capabilities that enable the secure, sovereign and interoperable exchange of data. Examples of these are Identity Providers, Participant Registries, Catalog Providers, Notarization Services.

Depending on the domain and use case, data sharing can have many different forms, such as API services, message streams, file exchange. It is not feasible for a generic data space infrastructure to directly support all specific data exchange modes. Therefore a data space consists of a control plane and a data plane, separating the generic core functionalities of a

data space from the specific exchange mechanisms. The control plane is responsible for functionalities that are independent of how data is exchanged:

- Data, services and offering catalog
- Contract negotiation
- Transfer management
- Identification, Authentication, Authorization and Trust

The separation of a data space into a control plane and a data plane allows the generic data space functionalities to be independent of how the data is exchanged. This allows for greater flexibility as opposed to having a strictly prescribed exchange protocol. Providing a general purpose solution to many different types of data sharing use cases and fostering interoperability between data spaces. It should be noted that the contract negotiation process by default is fully automatic, very basic and abstracted away from the user.

The data plane contains the implementation for the specific modes of exchange. Multiple data planes may be in use at once, each representing a different mode of exchange. It is the responsibility of the control plane to invoke the correct data plane, based on a request by another participant.

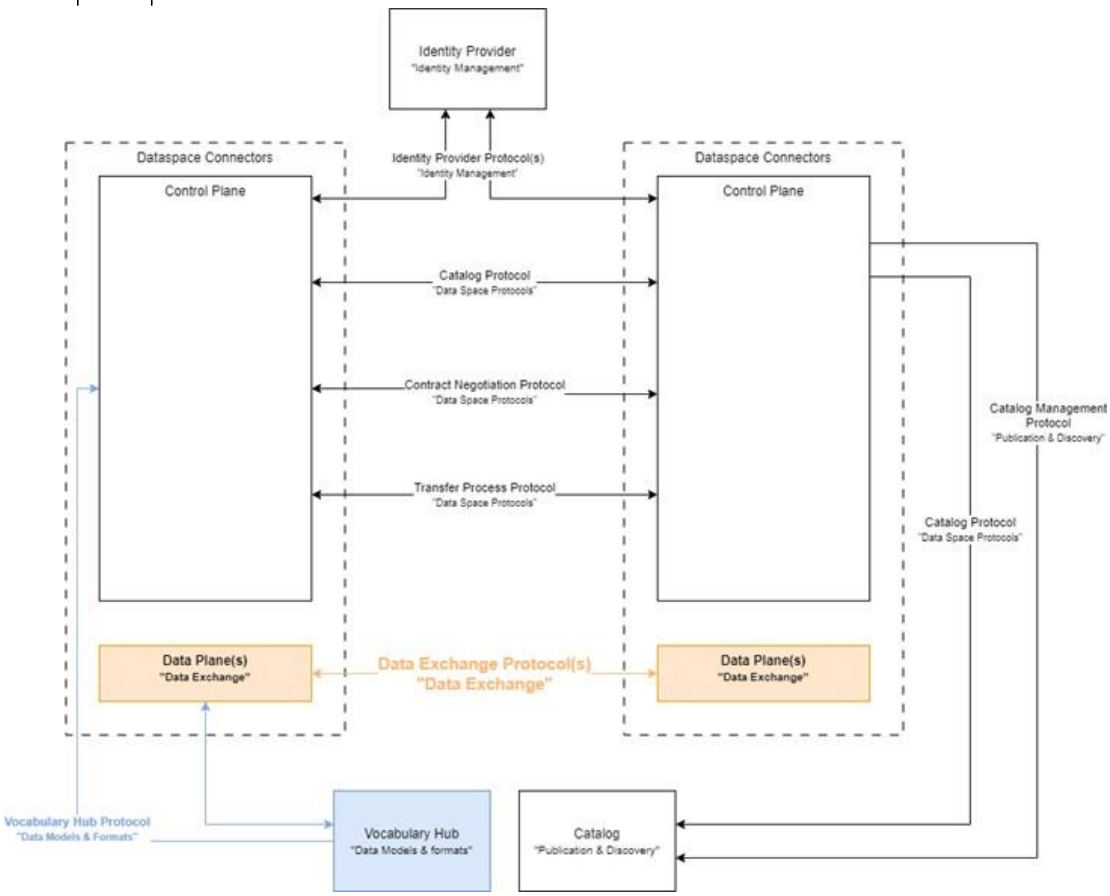

Figure 8.1 Example data space configuration with, depicting the control plane/data plane separation

8.2 Data Space Standards

A data space can use any suitable standard for the control plane. Particularly when it comes to building a trust framework several existing standards exist to choose from. Currently, there are two control plane standards specifically for data spaces that are worth noting, both are managed by the International Data Spaces Association (IDSA) [15]:

- Data Space Protocol[7]: a protocol for data space control planes it describes how:
  - How participants can exchange information on services
  - How a contract can be negotiated between participants that dictates the terms of service usage
  - How a transfer session is managed between participants
- Decentralized claims protocol (DCP)[8]: a protocol that describes how to create a trust framework using decentralized identities and verifiable credentials. It describes how:
  - Tokens can be issued and verified using a decentralized identity[21], i.e. a system without the need for a central entity creating and asserting identities.
  - Verifiable credentials [22] can be issued to participants as proof of some attribute asserted by the issuer
  - Verifiable presentations[23] are exchanged to prove attributes to other participants

Both of these standards are currently maturing and are being used in operational data spaces, in particular Catena-X [23]. The IDSA is currently in the process of recognizing both as an international data sharing standard. The Data Space Protocol leverages two important standards as part of its specification:

- DCAT: A standard for describing data and service catalogs can be described [17]
- ODRL: A standard for describing access and usage policies can be described [24]

The Data Space Protocol (DSP) and the Decentralized Claims Protocol (DCP) are complementary standards. A data space can choose to use the DSP with a centralized trust protocol based on OpenID Connect instead of the DCP. The Eclipse Data Space Components (EDC)[4] currently serves as the primary reference implementation for these standards. EDC offers an extensive component library that simplifies the integration of these standards into data space components. It is utilized by Tractus-X[25], the connector software stack of Catena-X, which is presently the largest operational data space in the EU. While the DCP is mostly implemented in the EDC, the issuance flow is still pending. As a result, issuing credentials requires a manual process until this functionality is fully implemented in the EDC, expected by the end of 2024. Despite this limitation, the current state of the DCP can be used as an operational trust framework.

Using data space technologies can be a powerful way forward to bring data sharing to the next level. It is particularly useful in larger scale networks where participants share data with a varying number of parties that are not known in advance and may have no formal relationship. It is important to recognize however that this added functionality comes at a cost of added overhead and complexity, drawbacks that may not always justify the benefits. Therefore it may not be as suitable for environments where parties exchange data with a smaller number of fixed parties.

9 Fit Gap Analysis

This chapter presents a comparative analysis of the current pilot implementation using Corda and an analogous architecture based on generic data space technology. The aim of this comparison is to identify gaps in the implementation, evaluate potential enhancements, and assess the practical implications for scalability, interoperability, and flexibility. As discussed in Section 8, a data space refers to a federated system designed for data sharing across participants, which may offer advantages in different contexts. To ensure a meaningful and fair comparison, the following assumptions about a potential data space implementation are made:

- The Data Space Protocol (DSP) is used to manage and govern data interactions.
- A trust framework in place
- A participant registry is in place to allow participants to discover other participants
- A central Data Space Authority governs and certifies the participants and services

9.1 Analysis method

To conduct a structured analysis, several comparison criteria have been established, derived from both the key features provided by Corda and the potential additional capabilities offered by generic data space technologies. These criteria include aspects such as scalability, data sovereignty, interoperability, and routing flexibility. Table 9.1 lists these criteria, which serve as the basis for evaluating the architectural differences between the two systems. While there is also a desire for customs to integrate Semantic Treehouse in the system, this was not considered for this analysis as this does not depend directly on the system. In each case, adapters have to be developed to use the Semantic Treehouse.

Table 9.1 Analysis Criteria

ID	Criterion	Source	Description
C1	Trust framework	FR-003, NFR-002	Providing the means to allow participants to setup trusted communication
C2	Discoverability of participants	FR-004	The ability to find other participants
C3	Discoverability of services	FR-004	The ability to find other services
C4	Data sovereignty capabilities	FR-001	The ability to dictate who can view your data and what they can do with it
C5	Dynamic Onboarding capabilities	FR-002	Whether onboarding of participants after setup is allowed
C6	Data exchange protocol	FR-001	Whether the system can be used to execute the data exchange
C7	Advanced routing capabilities	FR-001	The ability to define advanced rules for routing data across the network
C8	No vendor lock-in	NFR-003	If the implementation is specific to a vendor
C9	Based on international data sharing standards	NFR-003	Whether a solution is based on international data sharing standards
C10	Non-repudiation		The identity of the sender and the exact time of an event can be verified.
C11	Backend interfacing	FR-005	Whether a solution allows easy integration with existing backend systems of a participant

9.2 Results

The outcome of the comparison based on the criteria in table 9.1 is summarized in table 9.2. The most significant gap identified is the lack of a built-in method for direct data exchange in generic data space technologies. Unlike Corda, which facilitates event distribution and routing using advanced Cordapp rules, generic data spaces only provide the framework for such exchanges but do not handle the actual transfer. Moreover, generic data spaces lack an automatic data distribution method, requiring custom solutions for managing event-driven exchanges.

Table 9.2 Comparison Matrix

ID	Criterion	FEDeRATED-CORDA	FEDeRATED-EDC
C1	Trust framework	✓	✓
C2	Discoverability of participants	✓	✓
C3	Discoverability of services	✗	✓
C4	Data sovereignty capabilities	✓	✓
C5	Dynamic onboarding capabilities	✓	✓
C6	Data exchange protocol	✓	New Distribution Orchestrator
C7	Advanced routing capabilities	✓	✗
C8	No vendor lock-in	✗	✓
C9	Based on international data sharing standards	✗	✓
C10	Non-repudiation	✓	✗
C11	Backend interfacing	✓	✓

To better visualize how an architecture using generic data space components would differ from the current Corda-based implementation (figure 6.1), figure 9.1 presents a generic example architecture using data space technology. Components highlighted in red are not part of the data space standard. A new component—The Distribution Orchestrator—is introduced to replicate the functionality currently handled by Corda.

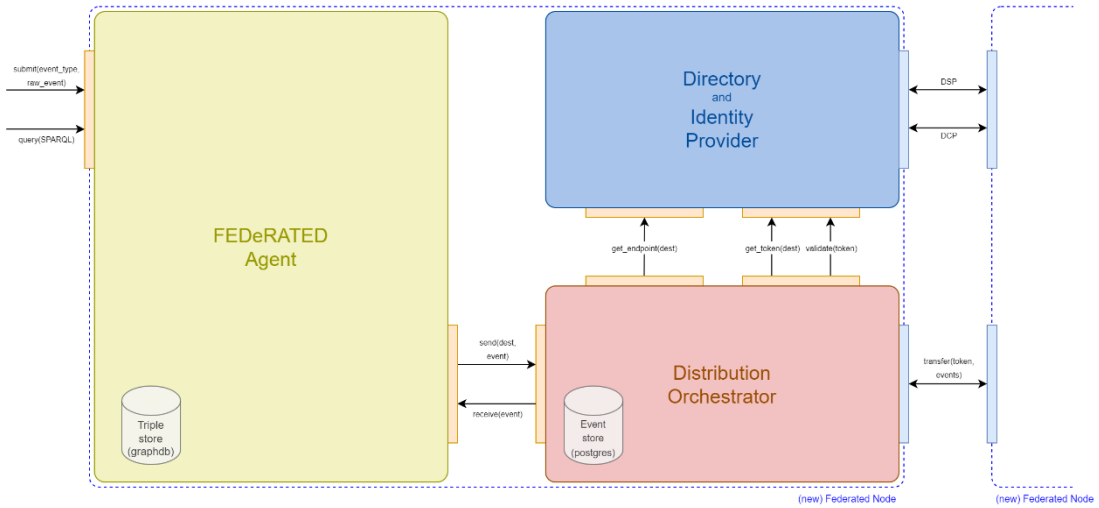

Figure 9.1 Proposed architecture based on generic data space technology

Distribution Orchestrator: This component handles incoming events from the API, determines their routing, and maps them to the appropriate data space participants. It locates the data services of the relevant recipients, negotiates contracts, and initiates data transfers. Once completed, it receives transfer details (such as endpoints and credentials) and pushes the events to the recipients.

9.3 Relation to BDI

The Basic Data Infrastructure Framework (BDI) is an infrastructure framework for controlled data sharing, supporting automated advanced information logistics in the physical economy [6]. The DIL programme is supported by the National Growth Fund and focuses on the accelerated development and application of the BDI (Basic data infrastructure) and on increasing the 'digital readiness' of (in particular) SME parties active in the logistics chain. Several living labs have been developed in order to do so.

On a conceptual level, based on the seven core principles of BDI, the differences between BDI and the Customs solution are fairly minor. Differences mainly consist of implementation choices. Unfortunately this still means that existing software components do not appear to be interoperable without significant work.

The extensive support for B2B interaction in the BDI is a clear difference with the use case implemented for Customs, which is based on B2G data sharing. In the Customs use case events can be (and are) shared between multiple participants, with Customs always being a recipient. In the publish-subscribe concept of BDI this is not automatically true. The focus of BDI on B2B, with a far more dynamic relationship between participants, presents some differences between BDI and Customs.

The trust framework in BDI is extensive, including not just identity management but also a reputation model, professional qualifications and a delegation mechanism. The trust framework is based on the OAuth2 protocol [26]. The EDC provides the Distributed Claims Protocol (DCP), which is a trust model using decentralized identities in the form of WebDIDs [27]. This module includes participant attributes (such as reputation, qualifications and membership) attestation using verifiable credentials. The current solution for Customs is basic, with identity management based on CORDA and trust being implied by being a participant in the CORDA network.

The event mechanism, including advanced routing capabilities and a mechanism for nonrepudiation, is specified to a very high level in BDI. The underlying concept is based on a subscription model, in which the recipient determines which events to subscribe to, while the current Customs solution is the opposite, with nodes capable of receiving all types of events and the sender determining which events are sent to each recipient.

Interfacing with existing legacy systems in BDI is based on commonly used technology to facilitate easy integration. The Customs solution provides these interfaces as well, requiring messages based on the FEDeRATED ontology and offering configurable translation services. Achieving semantic interoperability with the BDI will require additional work to translate between message models. Interoperability, particularly regarding event distribution, with BDI seems quite possible and is expected to be easier in the data spaces approach compared to the implementation using Corda. Although there are similarities between the trust mechanisms and options for interoperability seem possible, interoperability on this aspect will require effort since the underlying techniques are quite different.

This leads to the following comparison table. With less hands-on experience with the BDI the contents are not as clear-cut as those of the comparison between the current solution and generic data space technology, which is why these results are not merged with the previous table.

Table 9.3 Comparison Matrix

ID	Criterion	FEDeRATED-EDC	BDI	Interoperability
C1	Trust framework	✓	✓	BDI uses iShare, EDC uses WebDIDs+VC. Interoperability has to be investigated.
C2	Discoverability of participants	✓	✓	FEDeRATED uses the Douane Catalog, BDI uses iShare Satellite. Interoperability seems possible but will require investigation.
C3	Discoverability of services	✓	✓	
C4	Data sovereignty capabilities	✓	✓	Less relevant on the level of FEDeRATED and BDI. EDC offers this on the level of a participant.
C5	Dynamic onboarding capabilities	✓	✓	Not relevant for interoperability between FEDeRATED and BDI. Onboarding is a local procedure.
C6	Data exchange protocol	✗	✗	FEDeRATED uses Store & Forward, BDI uses Publish & Subscribe. Interoperability seems possible but will require effort.
C7	Advanced routing capabilities	✗	✗	Not relevant from perspective of interoperability
C8	No vendor lock-in	✓	✓	
C9	Based on international data sharing standards	✓	✓	In principle interoperable although care should be taken that the right design decisions are taken.
C10	Non-repudiation	✗	✓	No Interoperability problem. This is a local function.
C11	Backend interfacing	✓	✓	No Interoperability problem. This is a local function.

In summary: the two main topics for interoperability are Trust Framework and Data Exchange Protocol (event distribution). Discoverability of participants and services needs investigation.

9.4 Discussion

The primary discrepancy between Corda and generic data space technology lies in the facilitation of data exchange and the ability of Corda to route data through advanced rules built into Cordapps. This is a key advantage of Corda, which offers a seamless, built-in mechanism for pushing events to designated recipients. In contrast, generic data space technologies require additional components, such as the Distribution Orchestrator to perform these tasks through the data plane. BDI has been implemented with several different event exchange mechanisms, based on a publish-subscribe paradigm.

This gap has important implementation implications, as it necessitates the development of custom modules for orchestration and data transfer. As we have not been able to find solutions (open source or otherwise) providing the 'push'-paradigm which is central to the FEDeRATED vision, in which the initiative for the data exchange lies completely with the sending node without requiring an initiating action by the recipient of the event, a custom component will be required in all of the examined scenarios in which Corda is replaced.

The flexibility of generic data space technology provides certain advantages. For instance, the lack of a predefined exchange protocol allows the selection of different exchange modes based on specific payload requirements. In scenarios with high-volume, low-latency data exchange, a pub/sub streaming solution could be selected, whereas simpler solutions like HTTP APIs may be more appropriate for low-volume, higher-latency exchanges.
In the Customs scenario a 'push'-based component can be implemented.

An additional gap is the availability of service lookup capabilities. Generic data space technologies offers more services then Corda in this regard. Being based on international data-sharing standards provides a significant advantage where interoperability with other data-sharing environments is a requirement. The data space approach also provides better capabilities for maintaining data sovereignty, allowing policies attached to data services to be revised at any time. In contrast, Corda embeds data access rules within Cordapps, meaning that any changes to data access require rewriting and redeploying the entire program on the Corda node.

Although it is not used in the current solution, Corda supports non repudiation by default, through the use of cryptographic hashes and a notary node. While the data space implementation does not support this by default, custom components can be added, should this be a desired feature. BDI can provide facilities for non-repudiation.

Additionally, where Corda is a closed-source solution, open-source implementations of data space technology are available, such as the Eclipse Data Space Components framework. This openness provides a further advantage in scenarios where flexibility, transparency, and broad adoption are essential.

Additional gaps may be present depending on the chosen implementation. For example, the EDC only supports the DCP[4] standard as a trust framework. While this standard is specifically designed for data spaces and is expected to become widely adopted, an alternative framework, such as one based on OpenID Connect, may still be preferred. In such cases, additional implementations will be required.

10 Conclusions

This analysis has shown that a migration of the FEDeRATED Node from the Corda to generic data space technologies is possible, solving the problems presented in the introduction. Since generic data space technology uses internationally recognized standards, it helps to prevent a vendor lock-in. In the migration to the EDC, the event distribution including the 'push mechanism' (central vision to Federated) and advanced routing rules offered by Corda need to be replaced by introducing a distribution orchestrator.

Such a solution would also greatly simplify technical interoperability with other data spaces in the future. The most important Data Space in the context of FEDeRATED is Basic Data Infrastructure (BDI). A high level analysis shows that the main topics to consider for interoperability are 1) trust and 2) data exchange (Store & Forward versus Publish and Subscribe). Interoperability of discoverability requires attention.

It is also important that in the configuration of the EDC, FEDeRATED and BDI do not make inconsistent choices that hinder interoperability.

11 References

[1] https://www.federatedplatforms.eu/
[2] https://corda.net/
[3] https://datainlogistics.org/bdi/
[4] https://projects.eclipse.org/projects/technology.edc
[5] https://azure.microsoft.com/
[6] https://www.semantic-treehouse.nl/
[7] https://docs.internationaldataspaces.org/ids-knowledgebase/dataspace-protocol
[8] https://projects.eclipse.org/proposals/eclipse-dataspace-decentralized-claims-protocol
[9] https://transport.ec.europa.eu/transport-themes/digital-transport-and-logistics-forum-dtlf_en
[10] https://www.federatedplatforms.eu/index.php/federated-semantic-interoperability
[11] https://digital-strategy.ec.europa.eu/en/policies/data-act
[12] https://digital-strategy.ec.europa.eu/en/policies/data-governance-act
[13] https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/europe-fit-digital-age/european-data-strategy_en
[14] https://dssc.eu/
[15] https://internationaldataspaces.org/
[16] https://github.com/RMLio/rmlmapper-java
[17] https://www.w3.org/TR/vocab-dcat-3/
[18] https://www.w3.org/TR/rdf-sparql-query/
[19] https://dssc.eu/page/knowledge-base
[20] https://dssc.eu/space/bv15e/766064046/Building+Block+Overview#Data-Spaces-Building-Blocks
[21] https://www.w3.org/TR/did-core/
[22] https://www.w3.org/TR/vc-data-model/
[23] https://catena-x.net/en/1
[24] https://www.w3.org/TR/odrl-model/
[25] https://eclipse-tractusx.github.io/
[26] https://auth0.com/intro-to-iam/what-is-oauth-2
[27] https://w3c-ccg.github.io/did-method-web/

ICT, Strategy & Policy

www.tno.nl

TNO innovation for life