Blockchain in rail transport:

Condition based maintenance of railway wagons

June 2022

Authors:

Wout Hofman

Tijmen Elfrink

Erik de Graaf

Vincent Koppen

Christian van Ommeren

Table of Contents

1 Introduction

1.1 Objective

The objective is to assess the feasibility of blockchain technology for goods transport by rail by identifying opportunities and assess potential solutions, applying the tools and technologies of SPARK! Living Labs. The feasibility is explored by selecting a use case of wagon inspection and repair. This use case is the basis for Condition Based Maintenance of railway wagons.

1.2 Background

This document is a deliverable of the Spark! Living Lab (https://sparklivinglab.nl). The objective of SPARK! is to stimulate the adoption of and to create a community for application of blockchain technology by Small and Medium sized Enterprises (SMEs) in supply and logistics. As such, SPARK! has explored various use cases in different application areas of supply and logistics.

One of the objectives of SPARK! is the development and application of a generic toolkit for configuring use case specific blockchain networks. The toolkit has been developed by the CEF (Connecting European Facilities) FEDeRATED Action (federatedplatforms.eu). It consists of infrastructural components, called the Basic Data sharing Infrastructure (BDI) Nodes, that are a combination of semantic technology (GraphDB) and blockchain technology (Corda). The BDI is to be developed according to the Dutch Digital Transport Strategy (DTS). It will be applied in this use case.

Since SPARK! is about the adoption of blockchain technology, the utilization of this technology will not be discussed for the use case. However, a single use case with limited number of stakeholders might not require blockchain technology. The assumption that is taken is large scale adoption of a blockchain network by users and various use cases, in rail and other modalities. A first use case like the one for rail can be an initiator to create this type of blockchain network. Having such a blockchain network can be a basis for innovation, for instance by developing dynamic planning, exchanging reserved paths, and predictive dynamic maintenance. These types of innovative applications are outside scope; each of them will have data requirements that need to be supported by the blockchain network.

Considering this background and the objective, the feasibility of applying the semantic model and the BDI nodes for a rail transport use case topic of this document.

1.3 Potential business case

The potential business case for the use case is relates to an increased demand of rail transport as part of the modal shift and optimization of infrastructure utilization, improved organisation, and reduction of delays. These three potential advantages are described in more detail.

The capacity of the rail transportation network has a hard limit determined by both the infrastructure and its users, the Railway Undertakings. Reforms to both the **infrastructure** and the **organisation** of rail transport can increase the throughput of transport via rail. Improving rail infrastructure is hardware oriented and a slow and costly process that has little to do with data sharing for optimization of capacity utilization by blockchain technology.

The current organisation of transport via rail poses limits to optimal utilize existing transport capacity. To be able to handle demands for bigger volumes of rail freight this organisation should be improved. Rail transport is currently organised via a rigid system in which train operators book timeslots in which they can drive on specific sections of track. These are developed by so-called path allocation mechanisms applied by rail infrastructure managers. It is difficult to changes these path allocation algorithms, since they impact the complete planning of railway utilization, both for passengers and freight.

However, a train can only start driving a section of track within its assigned timeslot. If a train is delayed it is possible that it will miss its path. Missing a path introduces an even bigger delay as slots are allocated well ahead of time and finding an alternative slot ad-hoc is difficult. To prevent big delays, train operators introduce safety margins by booking additional paths, effectively extending the latest time at which they can start their trip. These safety margins take up paths that are then unusable by other trains, potentially of other Railway Undertakings, and thus decrease the throughput of the rail infrastructure. Potential improvements can be made by releasing all reserved paths that will not be utilized. These paths can then potentially be used by other Railway Undertakings. There is also a competitive aspect to exchanging paths.

In the current organization of rail transport the incidence of delay inducing events is so high that Railway Undertakings (RUs) are willing to pay additional path reservation fees for each trip to reduce the potential effect of delay inducing events. If the incidence of delay inducing events were lower, the necessity of safety margins would decrease, which would free up reserved paths, which would in turn increase the total throughput of transport via rail.

The incidence of some delay inducing events is difficult to decrease (e.g. weather conditions, infrastructure (component) failures, or accidents), while other events can be prevented by increasing visibility and collaboration. In this document we will use a use case to investigate some of these delay-inducing events, when they can occur, and their impact on rail transport.

Condition-Based Maintenance (CBM) of railway wagons based on (sensor)data is expected to reduce costs and improve process efficiency. There is already much research into CBM, also for other assets like aircrafts. Expectations of savings are high. This document will indicate what data can be available for CBM. CBM can be offered to various stakeholders as a service for different purposes, e.g. a railway wagon keeper to improve its maintenance and a train operator to improve checking a train before departure and planning maintenance during transport. Depending on the CBM service consumer, data must be accessed of different stakeholders. Indications will be given in this document.

1.4 Reading guide

The reader will be provided with:

- The specifics of the rail wagon inspection use case
- Technical solutions to the use case: Blockchain versus centralized platform
- How this use case could be implemented as a FEDeRATED PoC

Additional information and context are provided in the Annex:

- An explanation about blockchain for laypeople
- Explanation of the Basic Data Infrastructure (BDI) for laypeople

1.5 Disclaimer

The content of this document is based on interviews with relevant stakeholders and has not yet been validated by them. It also includes output of any previous projects in which TNO participated, like the H2020 SmartRail project.

The basis for development of a data sharing infrastructure for railway is the Basic Data Infrastructure (BDI) developed by the Dutch Government as part of the FEDeRATED project. The BDI includes blockchain technology. Details can already be found at www.federatedplatforms.eu. It includes a semantic model for data sharing in logistics and an (draft) architecture. Any relevant details are copied and summarized in this document; we refer to the original documents for more precise documentation.

2 Guiding principles

Various communities have developed guiding principles for data sharing, e.g. the Digital Transport and Logistics Forum (DTLF) which is an expert group raised and chaired by EC DG Move, the CEF funded FEDeRATED Action, and the International Data Space Association (IDSA). The ones of FEDeRATED as listed in the Interim Master Plan are the most detailed. These refine the ones of DTLF.

2.1 General principles

The objective of DTLF is to create an open and neutral data sharing infrastructure that is accessible to all stakeholder (level playing field, which means that also SMEs should be able to join the architecture). The infrastructure consists of a so-called 'federated network of platforms' which implies that installed base of platforms and investments done by stakeholders are safeguarded. They need to become interoperable somehow.

In this context, a set of technology independent agreements needs to be reached that enable organizations to share data. These agreements consist of:

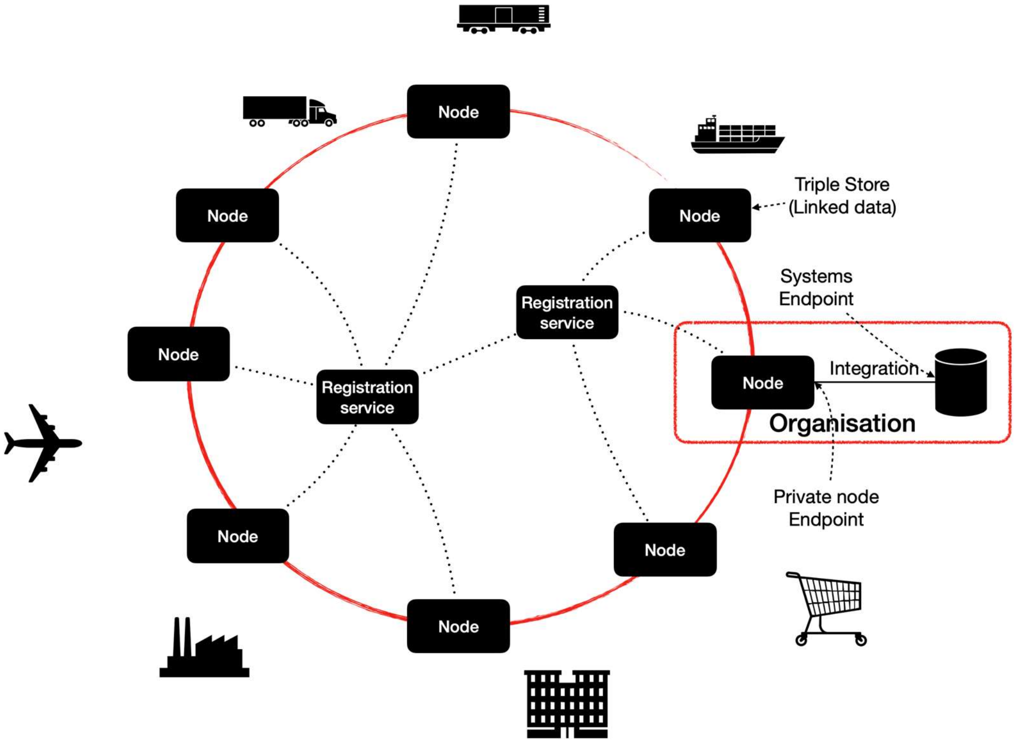
- **Common language** – both the semantics and their representation for data sharing should be clearly specified to automatically share and process data by different IT systems, supported by IT services (APIs – Application Programming Interfaces). This is called the semantic model. The APIs enable business process collaboration.
- **Identity and Authentication** – each organization should have a unique identity that is issued by a certified identity provider and can be authenticated. Multiple identification domains may have to be specified, each based on its certification mechanism supported by an identity broker. eIDAS (electronic IDentification, Authentication, and trust Services) is an example where the EU Member States have implemented an agreed certification mechanism for B2G data sharing, both for users and IT systems. Open standards should be applied, in combination with APIs (e.g. OAUTH2.0 and REST API identity tokens).
- **Data sovereignty** – each enterprise should be able to control its data sharing, compliant with any restrictions (e.g. GDPR) and regulations (e.g. UCC and eFTI). This is part of access control.
- **Discoverability** – for inclusiveness and optimization it should be possible to discover commercial information, business services, available logistics capacity, and the past (e.g. a trace or container track), present, and future (e.g. a planned flight, itinerary, or voyage with available capacity) state of supply and logistics chains in networks. State changes are shared via events that support business collaboration. Additionally, information services like weather conditions must be findable.
- **Data sharing solution** – the actual means for reliable and secure transfer of data, including facilities supporting non-repudiation (log and audit trail). Such data sharing solutions provide connectivity, based on (secure) protocols like Transport Link Security (TLS). Most implementations of these standard protocols have an additional layer to facilitate data sharing, where data is the payload with an envelop containing control information required for processing the payload at reception.

These aspects are currently under development by DTLF, resulting in a proposed governance structure. IDSA provides a data sharing solution combined with an approach for non-repudiation based on a Clearing House and a broker acting as means to discover business services. Blockchain technology inherently supports non-repudiation and can, additionally, also support data integrity.

Corda provides an peer-to-peer data sharing solution with temporary data storage in a Corda node, supported by an underlying notary network for non-repudiation and data integrity.

The common language has already been specified. In the Dutch Living Lab, the **Basic Data sharing Infrastructure** (BDI) is developed, where nodes constitute a network. These nodes fully implement the common language. Any type of query can be formulated on this model, for instance 'container track' or 'traffic density'. Such a large degree of freedom is too complex to handle by most logistics stakeholders, especially SMEs. Thus, predefined settings must be provided like for eFTI, eCMR, and eB/L.

A **BDI node**, which is part of the network, consists of a Corda node for data sharing and a triple store that supports discoverability. The triple store contains all events that are shared in a peer-to-peer setting (this means that BDI nodes all contain different data). Each participant implements a BDI node (or its required functionality), thus constructing the BDI. The BDI can have more than one registration service at which one registers its node. The current version of the BDI utilizes the Corda network management service for registering the Corda node as a BDI component to the network.



Industry associations and regulatory bodies can play an important role for predefined settings. They are key stakeholders in the governance. This will allow large scale application of solutions, whilst organizations are still able to innovate. These governance bodies and first movers implementing the solution will drive market acceptance.

The solution can be implemented like a regular change project where new sets of APIs are introduced, but it can also be a paradigm shift. Complete implementation of the semantic model (and any changes) supported by predefined settings and tools to formulate queries does not require standardization like we currently know (for instance of APIs and messages). The infrastructure is open, neutral, and fully distributed. Data sharing platforms are not required; value added functions to support data sharing (e.g. data transformation, data analytics) and logistics (e.g. dynamic chain planning) are required. These will contribute to synchromodality and a seamless goods flow.

Such an infrastructure is expected to drive innovation and enable all types of new services and business models, based on data sovereignty. Various views of supply and logistics can be given, for instance for maintenance and repair and bundling of goods flows.

2.2 Detailed principles

FEDeRATED provides detailed principles that might be applicable to this proposal. These are shown in the next table. The last column shows the applicability of a particular principle to this use case.

FEDeRATED LEADING PRINCIPLES			
Principle	**No.**	**Description**	**Applicability**
Level Playing Field	1	All supply chain operators and public authorities involved in freight transport and logistics have to be able to participate.	Y
Electronic/digital format	2	The information is to be encoded digitally, using a revisable structured format.	Y
	Principle 2 refers to technical interoperability. The information is to be encoded digitally, using a revisable structured format, which can be used directly for storage, and processing by computers, such a structured format for digitally encoded messages that can be transformed into for instance PDF.[1]		
Compliance with existing rules	3	Data sharing must be compliant to existing legislation (e.g. GDPR) and privately agreed rules.	Y
	Principle 3 refers to legal interoperability		
Business service	4	Each participant has to formulate the business service(s) it provides (service provider) or requires (customer).	(not yet)
	Principle 4 addresses organizational interoperability for enterprises		
Business relations	5	Trust between enterprises is primarily driven by their real work relationships.	Y
	E.g. an enterprise can trust a (known) service provider, but not necessarily another one with whom that enterprise did not do business		
Supply and logistics chains	6	The business relations between participants are shown according to their outsourcing hierarchy from the perspective of for instance a shipper and/or consignee.	Y
Data requirements of enterprises	7	Business services and commercial mechanisms supporting negotiation between a customer and service provider specify the data that they will share.	Y
	Principle 7 contributes to semantic interoperability.		
Data requirements established by an authority	8	Data requirements set by an authority are related to the legislative basis afforded to that authority.	-
	Principle 8 refers to legal interoperability and organizational interoperability for authorities		
Data processing	9	Any organization can specify its internal processing.	Y
		E.g. outsourcing strategy (enterprises) or governance of cargo flows by risk assessment (authorities like customs).	

[1] XML, EDIFACT, JSON(-LD), and RDF(s) are supported. Mail attached files, i.e. PDF, Excel, Access, and JPEG, are not supported

FEDeRATED LEADING PRINCIPLES			
Principle	**No.**	**Description**	**Applicability**
Fit for purpose	10	Public authorities that access enterprise data require a legal basis to refer to.	-
	Principle 10 refers to legal- and organizational interoperability		
Publication of data requirements	11	Public authorities publish their data requirements in a machine-readable form.	-
	Principle 11 iterates that public authorities publish these data requirements to enable rapid and consistent implementation of these requirements by enterprises, thus reducing errors and supporting rapid changes.		
Business Service Discovery	12	Business services of all enterprises are discoverable according to harmonized search criteria	(not yet relevant)
Data as proof	13	A public authority or enterprise must be able to proof compliance or non-compliance with data.	Y
	Principle 13 stipulates data needs to be stored in a non-repudiated manner to allow such proof.		
Authorities providing data (authority services)	14	Public authorities can share their data with enterprises for policy reasons within a legal framework	-
	Principle 14 refers to legal interoperability and organizational interoperability for authorities		
Push/pull	15	A legally allowed data sharing mechanism allow in case of: • a push, data to be duplicated by enterprises to authorities; • a pull, data being made accessible to authorities.	-
	Principle 15 is part of technical interoperability. In case a regulation does not prescribe a mechanism, the pull mechanism is preferred to prevent unnecessary data duplications and thus errors. A reporting data set is only virtual: it is not stored separately but extracted from all other data sets based on a data pull by an authority. The eMSW data set consists of additional data sets like passengers and waste, which is for further development. However, the eMSW data set will be made available in a similar manner		
Publish/subscribe	16	An organization must have the ability to subscribe to any relevant new data in accordance with fit for purpose (public authority) or a commercial relationship (enterprise).	Y
	Principle 16 is part of technical interoperability. A data provider issues a unique link to the relevant data and will distribute data when it becomes available.		
Combining data requirements	17	Whenever a public authority is responsible for governance of more than one regulation, the data requirements of those regulations will be combined into one data set.	-
	Principle 17 refers to legal interoperability and organizational interoperability for authorities		
Identification of organizations	18	Each organization is able to identify itself uniquely according agreed attestations with transparent validation	Y

FEDeRATED LEADING PRINCIPLES			
Principle	**No.**	**Description**	**Applicability**
		processes of these attestations (e.g. Chamber of Commerce Registration, AEO certificate)	
Identification of users	19	Persons that act on behalf of an organization can identify themselves as such and should be known and employed or delegated by that organization	Y
User capabilities	20	The capabilities. i.e. the actions that may be performed, of an identified user are transparent to all other relevant users/organizations	Y (partly)
Data sensitivity	21	Sensitive data should not be accessible or changed by unauthorized users or organizations.	Y
	Principle 21 implies access to data that is stored or shared via some solution/platform. is applicable to for instance commercial sensitive data.		
Metadata of data sharing	22	Any metadata specifying which data is accessed or shared between any two enterprises is not accessible by unauthorised users or organizations.	Y
	Principe 22 addresses that business patterns can be derived from data shared between any two enterprises and should be hidden from third – non authorised - parties. It implies that metadata of data sharing between public authorities and enterprises is open data.		
Identification of systems	23	IT systems of an organization that support the roles data provider and -receiver, are uniquely identifiable	Y
Data sharing policy	24	A common policy or agreement specifies the use and reuse of data as well as the manner in which it is stored or removed.	Y
Data sovereignty	25	A data owner determines the data it will share and retains full rights and controls over this data	Y
Data at source	26	Single sharing of links, multiple (controlled) access to data	Y
	Principle 26 indicates that data should be stored at the source to prevent any duplication and potential errors, unless prescribed by a regulation or agreed upon by two organizations that share the data. To have data at the source, these organizations only share links to that data.		
Data sets	27	The data sets of which links can be shared is given by the semantic model	Y
	Principle 27 addresses semantic interoperability.		
Baseline standards	28	Use of baseline standard(s) that provide all common terminology, data formats, code values, etc. that can be re-used for implementation of the FEDeRATED models.	Y
	Principle 28 on baseline standards address for instance code values like ISO country codes, ISO standards for date/time formats and terminology with formats like specified in the UN CEFACT Core Component List (see chapter 7)		
Data timestamps	29	An event for sharing milestones has its own timestamp that can differ from the timestamp of a milestone.	Y
	Principle 29 identifies the need for difference between these timestamps to be small in the context of process synchronization		

FEDeRATED LEADING PRINCIPLES			
Principle	No.	Description	Applicability
Unique identifier(s) of data (sets)	30	Unique identifiers are used to create and share links of relevant data sets between any two enterprises.	Y
	Principle 30 identifies that unique identifiers might differ from identifiers used in the real-world, e.g. a container has a unique container number and can have a unique link for data sharing.		
Data sharing solution	31	Organizations select a solution of choice for data sharing with others (platform, peer-to-peer)	-
Federation	32	Organizations are able to share or access data with others	Y
Data validation	33	Data is either validated by a data provider or a – receiver against data sharing specifications (e.g. XSD).	Y
	Principle 33 identifies that a data receiver will always receive an indication of validation to prevent any double validation. Data validation is on completeness and correctness.		
Data Exchange integrity	34	Accuracy and consistency of data over its entire lifecycle is required	Y
	Principle 34 identifies that the fundamental elements of trust in data are to ensure data audits and non-repudiation hitch. Data delivery must also be guaranteed to ensure trustworthy data exchange		
Historical data	35	Historical data sets are stored for optimizing business processes (public authorities and enterprises), based on legal requirements (e.g. archiving),	Y
	Principle 35 iterates that data can also be used to support Research & Development and statistics.		
Logging and audit trail	36	Organizations store a (shared) immutable log and audit trail of the data they have shared.	Y
Monitoring	37	Each organization is able to trace with whom and at what time particular data has been accessed/shared with any other organization.	Y

2.3 Data sharing infrastructure and services utilizing the infrastructure

The objective is to create a data sharing infrastructure constituting the BDI, in this case based on blockchain technology. It is an infrastructure for sharing data between a data holder and – user with capabilities like:

- **Data sovereignty** implemented by
  - Peer-to-peer data sharing of events
  - Data pull based on links to data received via the events
  - Access control configured by a data holder, compliant with regulations
  - Support of Identification, Authentication, and Authorisation (IAA)
- Decoupling of **search** from data sources. A data user can develop its own query/Graphical User Interface (GUI) based on the semantic model and the data stored by its BDI node. A data holder needs to provide data (events and pull) according to the semantic model.
- **Data quality** validation at different levels
  - Each interaction is validated against its specification by the semantic model
  - The sequence of interactions is validated in the context of a business service provided (under development).

- **Event distribution** based on agreed rules. Events can be automatic distributed from a service provider to a customer or the next leg in a logistics chain, utilizing specific rules. For instance, detection of a malfunctioning railway wagon by an IM can be automatically distributed to the RU, the Railway Wagon Keeper, and the Railway wagon maintenance (see further). In a similar way, delays with updates of ETA's can be distributed.
- **Query federation** to enable access to a data source (data provenance). For instance, all data of a railway wagon is stored by its keeper and the usage is stored by an RU and made accessible to the keeper via a link. Cargo details are stored by the production plant. Links provided to the shipper, the RU, and the IM based on the commercial relations (see next chapter).

Any **(data analytics) service** like an ETA predictor or a dynamic route planner provided by a third party can utilize the data shared by the BDI. Such a service will operate on behalf of a stakeholder acting as user of the BDI. The service needs to specify its data requirements and its output, e.g. an ETA with a particular quality. These types of services are not part of the BDI.

For instance, one can imagine a wagon maintenance service that utilizes wagon data of a Railway Wagon Keeper and data with respect to its utilization provided by an RU to the keeper. The service can be provided to a keeper and an RU. In case an RU utilizes the service, it may require data of other RUs that have used the same wagon. This data can be provided by the keeper. Query federation or event distribution is required to access the data source and pull the data to provide the service.

3 Wagon inspection use case

This section describes the use case by analysing the wagon inspection process, identify data that can shared and map it to the semantic model.

3.1 Description of the use case

Nedmag produces chemical substances for a wide range of products. For many of their products they use dolime (calcium & magnesium oxide) which is mined near Hermalle (Belgium). This dolime is transported via rail to Veendam (Netherlands). The process is simple but nonetheless prone to delay inducing events. Transports of dolime has four relevant and distinct steps:

1. A train loads dolime at Hermalle
2. The train drives to Veendam
3. The dolime is unloaded into storage bins in Veendam
4. The train drives back empty to Hermalle

Nedmag orders *three trainloads* per week of its Railway Undertaking (RU) Lineas meaning that step 1 though 4 are repeated three times per week. Trains are booked on specific timeslots and they cannot start their journey outside of this timeslot. This means that if the train is drastically delayed during the trip from Hermalle to Veendam (2), the return trip (4) cannot be started on time. Thus, the train cannot get back to Hermalle to pick up the next delivery of dolime, defaulting on the necessary operating frequency of thrice a week.

Each train can carry at most 1.400 T of dolime. This means that at maximum Nedmag expects 4.200 T of dolime from Hermalle each week. There are two main factors that can decrease the amount of dolime that is delivered to Nedmag:

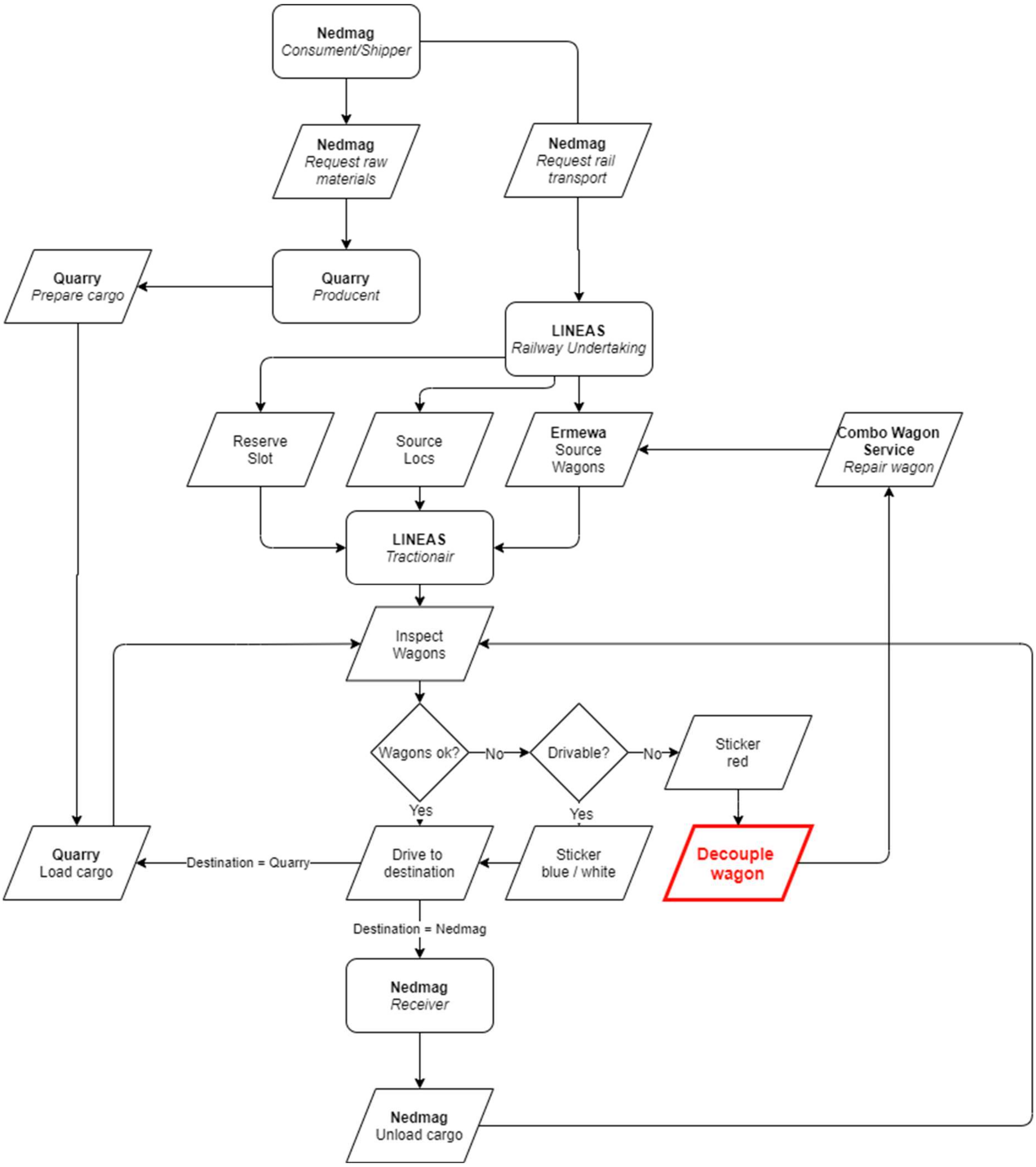
1. Not enough wagons are available to transport a full trainload of 1.400 T reducing the load of a single transport (e.g. only 1.000 T instead of 1.400 T is transported).
2. Delays in transport from Hermalle to Veendam causes the **next** transport from Hermalle to Veendam to miss its timeslot and thus Veendam has 1.400 T less dolime than expected.

3.1.1 Delay by wagon failure

Nedmag indicates that wagon failure is a major cause for loss of transport capacity. Due to safety regulations wagons must be checked prior to every transport. If the status of a wagon is not up to par with regulations it must be excluded from transport. This necessary but timely process can be a *direct* source of reduction of transport capacity (the first type described above under 1) or *indirect* reduction (the second type described above under 2). *Direct reduction* is caused by a lack of suitable wagons that can be used for transport. *Indirect reduction* is caused when the additional time it takes to remove faulty wagons from the train causes the transport timeslot to be missed. If this happens the entire transport must be rescheduled.

3.1.2 Stakeholders, roles, and actions

Transport over rail requires multiple stakeholders which facilitate (sub)processes such as carrying responsibility for cargo handovers, provide train-, wagon- and rail-capacity and perform repair or maintenance tasks. As responsibilities can shift in each step of the logistic process, access to information regarding the actions performed by the latest involved party is required multiple times by different parties during the completion of a transport process. Relevant stakeholders in the Nedmag use case are displayed in the figure below and elaborated on in this section.



3.1.2.1 Nedmag

Nedmag produces chemical substances for a wide range of products. For many of their products they use dolime, which is mined near Hermalle (Belgium). This dolime is transported via rail to Veendam (Netherlands).

3.1.2.2 Lineas

Lineas offers rail transportation services and is responsible for providing an available train and tractionair (train operator) as a service, and for executing wagon inspections. As RU and train operator, Lineas provides the driver for the train.

3.1.2.3 Combo Wagon Care

Combo Wagon Care facilitates the repair and maintenance of railway wagons. For Combo Wagon Care it is essential to have access to information regarding the condition and service history of railway wagons but Combo is also required to exchange data (for instance regarding wagon availability) with railway undertakings in order to efficiently cooperate with each other.

3.1.2.4 Ermewa

Ermewa owns a fleet of railway cargo wagons which are offered in a lease construction to parties such as Lineas. Ermewa does not maintain or repair wagons by themselves but relies on Combo Wagon Care to facilitate this service.

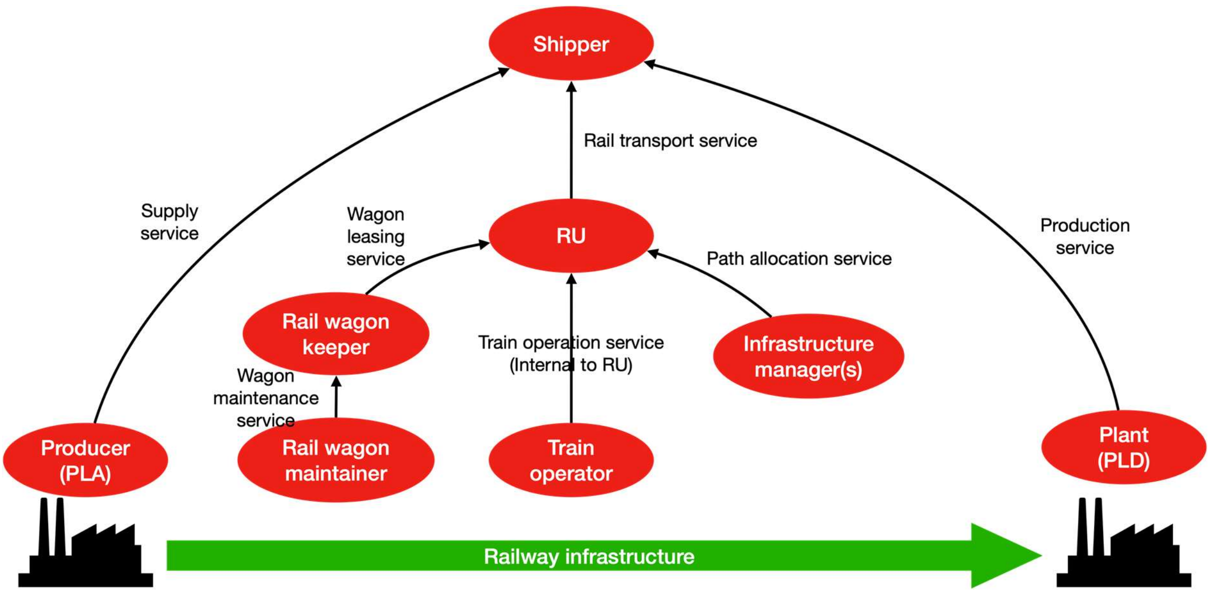
3.1.3 Roles

We identify a defined set of involved actors in this use case, each fulfilling an essential role. These roles are:

Stakeholder	**Role**	**Description**
Nedmag	Shipper/consignee	Producing chemicals, ordering transportation Receiving + processing dolime
Lineas	Railway Undertaking	Organising railway transportation: Contracting locomotives and wagons, and reserving infrastructure slots.
Lineas	Train operator	The person operating a train on behalf of Lineas. The person is qualified and responsible for sourcing and driving the locomotive and inspecting the wagons.
Emerwa	Rail wagon keeper	Owner of the wagons. The wagons are provided via Lineas.
Combo Wagon Services	Rail wagon maintainer	Organisation that maintains and repairs wagons on behalf of Emerwa.
ProRail (NL)	Infrastructure manager	Organisation that provides paths to an RU on the rail infrastructure, monitors positions of trains (locomotive + wagon), and is responsible for safety on the rail infrastructure. It implies that an RU has to provide details on the state of a train (its wagons and traction) and details of the cargo (e.g. dangerous cargo).

3.1.4 Transaction tree

These roles and their commercial relations are depicted by the following transaction tree. Nedmag, acting as shipper and consignee, orders rail transport from a supplier (e.g. Hermalle) to one of its production plants (Veendam). Two rail infrastructure managers (IM) are involved, that of Belgium and ProRail of the Netherlands. A path allocation service is provided by the Belgium IM, in coordination with ProRail. This may cause some delay at the border, since allocated paths might not be matching exactly (in time). Each path consists thus of one or more stretches or transport legs. Handover at a border needs to take place between two stretches.



Each arrow in the previous figure shows the business service provided by a role. For instance, a Rail Wagon Keeper provides a wagon leasing service to an RU. The Train Operation service is an internal service; it provides details to the Train Operator for driving its paths from producer (Place of Acceptance – PLA) to the plant for destination (Place of Delivery – PLD). The PLD is owned by the shipper, implying that the production service is also an internal service.

The transaction tree represents the responsibilities (and liabilities) of roles involved in the use case. It implies for instance that when a train operator detects a wagon that needs repair at inspection, it is reported to the RU that provides this information to the Rail Wagon Keeper. The latter must inform the Rail Wagon Maintainer that can provide a maintenance plan (time and location) for the wagon.

Basically, a Rail Wagon Keeper needs to provide material that can be applied by the RU. The Rail Wagon Keeper can plan maintenance based on usage and inspection details of a wagon. The latter requires better information provided by a Train Operator, who will most probably not have the time available for providing it.

An RU is responsible for using material in accordance with the safety requirements of an IM. When ordering a path, an RU implicitly complies to these requirements. An IM will also (regularly) inspect whether the material (wagons and traction) complies with these requirements. In case an IM detects a malfunction of a wagon, it will probably decouple the wagon from the train and inform the RU. The RU in its turn will inform the Rail Wagon keeper that will organize maintenance.

Any delays during transport will be reported by an IM to the RU. The RU can inform the shipper, including an improved ETA (Estimated Time of Arrival). This ETA will be passed on to the plant (PLD) to enable the arrival of a train.

The previous figure implies that for each train, a path allocation service will be called with one or more IMs. This is not the case. RUs have a number of paths assigned to them for a period of time, e.g. yearly they may have a path on a week day between for instance Hermalle and Veendam. The use of this path will be shared between RU and IM as will be shown next.

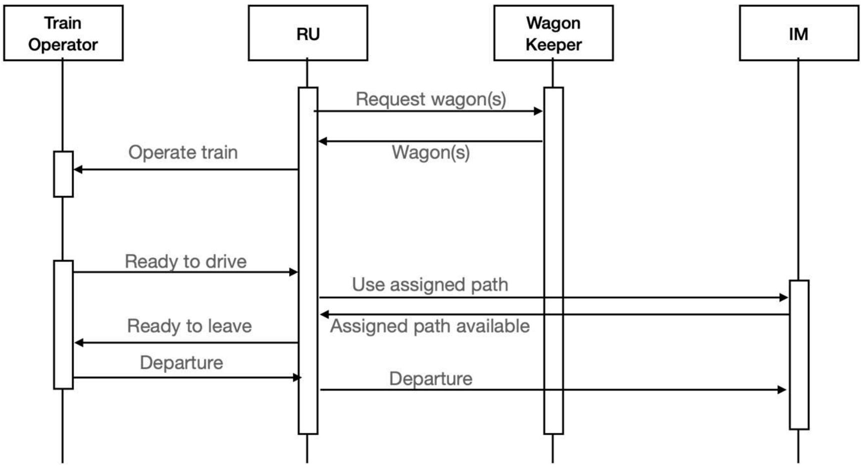
The previous figure has alternatives. For instance, if a shipper leases its own wagons, the wagon leasing service is provided by the Rail Wagon Keeper to the shipper. It means that an incident report is made available via an RU to a shipper, that makes it available to the Wagon Keeper. Another alternative might be that an RU organizes maintenance itself. The wagon maintenance service is thus offered to an RU. This might be part of the contract between an RU and a Rail Wagon Keeper.

3.1.5 Event distribution rules

Operational, events can be generated (as will be shown next) that represent the status of a wagon and the progress of a train from its departure to its destination. Distribution of events is based on two rules:

1. The existence of a commercial relation – any role makes an event that is relevant to another role available to that other role. This can be a service provider like a Train Operator providing a link to an incident report to an RU, but also an RU providing that link to an IM and Rail Wagon Keeper.
2. The type of business service – and agreement – this specifies the relevancy of distribution of events. For instance, a path allocation service (probably) requires a 'green' inspection report for all wagons in a train (or 'orange' at the most, see next section).

Applying these rules leads to for instance the following sequence diagram (not all roles are visualized).



The figure shows that an RU starts by requesting wagons. After wagon availability, a Train Operator is assigned to a train and receives its instructions. Inspection of the wagons after train composition leads to a request to drive an available path. This request is forwarded to the IM that indicates the path is still available. The figure continues of course after departure of a train.

The various interactions shown in the previous figure will be represented as events. These events relate to a physical operation of a service according to an interaction sequence. The interaction sequence basically consists of three types of interactions, in the context of a framework contract that we assume exists between stakeholders:

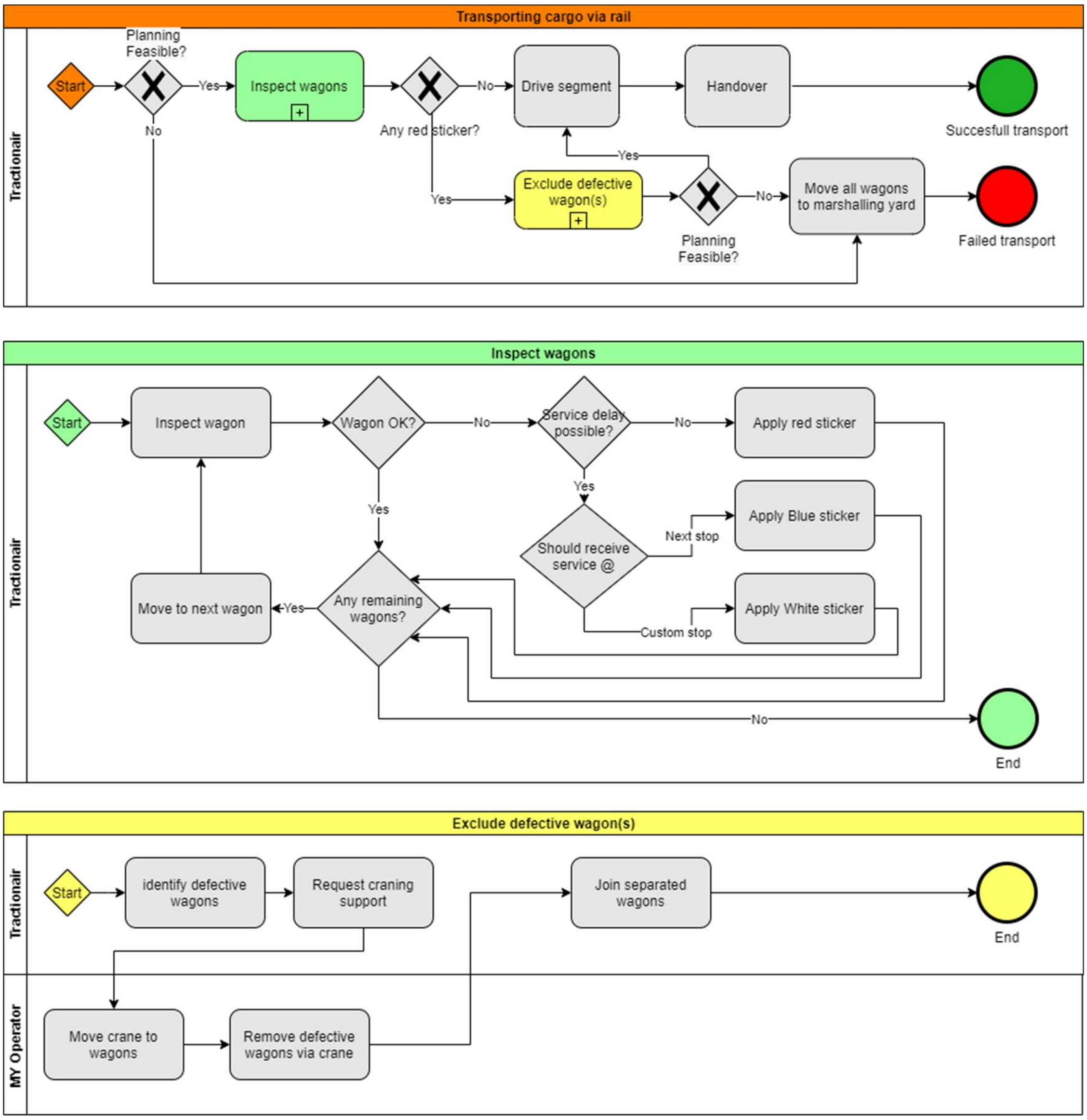
- Order – the expectation of a customer for delivery of a business services. This results in for instance the expected time in an event. Depending on the business service, the order event relates to one or more location, e.g. transport is always between two locations.
- Plan – the planning of the execution. A planning event always contains the estimated time.
- Report – this is the actual reporting of the operation. A departure event is an example, it contains the actual time at which departure took place.

These three interaction types must be considered in the context of a business service. It results in for instance 'Operate train' that is in fact calling the train operation service with an order interaction.

The planning is indicated as ‘ready to drive’. It contains train composition events and a link to an inspection report of each wagon in the train. These data event types are presented later in this document.

3.1.6 Wagon inspection in more detail

The process of wagon inspection by a Train Operator is analyzed in more detail without specifying each individual activity in the process. All the process steps such as sourcing wagons, sourcing a locomotive, and slot reserving are assumed to be taken care of. A business process model is visualized in the figure below with identification of stakeholder roles that are responsible for a particular part of the process. Additional explanation is provided below the figure.



The ‘inspect wagon’ – and the ‘exclude defective wagon(s)’ activity of the top diagram of the figure, ‘transporting cargo via rail’, are decomposed further.

The **Transporting cargo via rail** process starts at the **orange start tag**. At this stage, the locomotives and wagons are connected into a train and cargo is already loaded into the wagons. If the planning is no longer feasible (i.e., the timeslot of the available path has expired due to delays in another trip) the train is moved to the marshalling yard. If the planning is still feasible the tractionair is legally obligated to inspect the status of the wagons before the train can take its path from the start location to the end location.

Wagon inspection is formalized in the activity called **Inspect wagons** starting with the **green start tag**. The tractionair walks alongside the train and inspects every wagon. If the wagon is in good condition, the tractionair moves to the next wagon. If a wagon is detected that requires service (for instance due to unacceptable wear to the wheels, locks, or the breaks) the tractionair must make decide if the wagon must be serviced now, or if maintenance can be delayed just a little longer. Delaying maintenance has the advantage that no additional delay is introduced due to wagon removal but the disadvantage that the wagon could break during transport which has potentially catastrophic consequences (such as train derailing). The tractionair applies a physical token (in the form of a sticker) to wagons that require maintenance. These stickers can be **red** (immediate maintenance is required), **blue** (maintenance is required next stop), or **white** (maintenance is required at custom specified stop). The tractionair evaluates every wagon in the train before moving to the next step.

The **yellow start tag** indicates the starting point for the **Exclude defective wagons** activity. At this stage it is known which wagons should be excluded as the inspection process is finished. Wagons that must be removed are craned out of the train. The train is reconnected, and the exclusion process is finished.

The time it takes to complete the **Transporting cargo via rail** process highly depends on how long it takes to execute the **Exclude defective wagons** process, which in turn is dependent on if the **Inspect wagons** yields defective wagons. If the train was on schedule and there are no defective wagons identified (I.e., no red stickers) the chance that the train is still on schedule is very high. However, *with each defective wagon the probability of the train missing the following time slot increases*, as excluding these wagons simply takes up time. As is, the status of a wagon is identified by the tractionair and logged via stickers.

3.2 Less wagon exclusions through increased supply chain visibility

The need to decrease ad-hoc exclusion is evident seeing that it causes both a *direct decrease* in transport capacity (loss of wagons) and an *indirect decrease* in transport capacity (missing paths).

In the current situation there is little information shared between Train Operator and the Railway Undertaking regarding the status of wagons. This lack of information makes it difficult to plan wagon service intervals. If it were widely known that a wagon is entering the last leg of its service interval, opportunities to service this wagon with minimal impact on rail transport process could be sought out.

The first step in making wagon service a more plannable affair is increasing the availability of up-to-date information regarding wagon status. This information must be accessible for the stakeholders in rail transport.

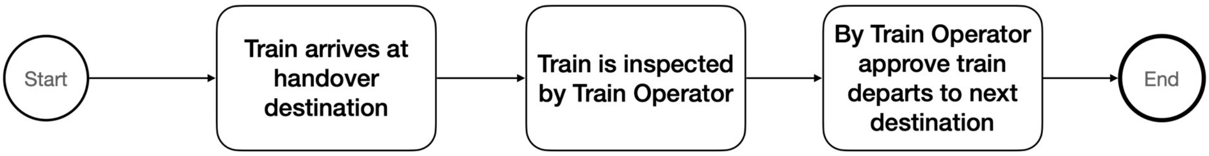
The incidence of ad-hoc exclusions could be decreased further by supporting the visual inspections that the Train Operators carry out with predicting maintenance horizons using sensor suites that monitor wagon status.

In the next section we identify the most notable events regarding the evaluation of wagon status.

3.3 Events generated during wagon inspection

To express each action or process step in forms of data, we make use of describing each part of a process as an event. An event is characterized by its nature of containing a transaction or action.

The process of inspecting a railway wagon has been split into three sub-processes as described in the paragraph above. However, in a simplified 'happy-flow', the process consists out of three elements shown in the figure below.


Start
Train arrives at handover destination
Train is inspected by Train Operator
By Train Operator approve train departs to next destination
End

The simplified process contains a set of recurring entities and actors which all have relational meaning between them. For instance, a train consists of a locomotive that is pulling a (or multiple) cargo wagon(s). However, each wagon is an entity on itself and has its own unique identifiers and properties. Some of the properties are easily perceivable by human actors but in many cases, properties exist out of a measurement or condition status. In this use case, Pharox plays an important role as a provider of sensors which can be attached to railway wagons.

This becomes useful in situations wherein it is required to keep track of changes in condition(s) or location(s). Every time a railway wagon is made to be part of a different train we would be able to determine which Train Operator inspected the wagon and therefore created the inspection report. Accompanied by information regarding the location and last carried cargo of that wagon, we could start determining characteristics or making conclusions out of the data that is generated.

In the compact flow-diagram above we distinguish three events which refer to wagon status information, of which many properties are described in the table below. An example value is provided for each property; in some cases it refers to 'UUID' which is a Universal Unique Identifier that refers to the source of the data.

Digital twin: Railway wagon		**Data holder: Wagon Keeper**
Property	**Description**	**Example value**
Has a unique identifier	An identifier value is bound to the physical railway wagon	ID: RW0001
Empty weight	Shows the empty weight of the railway wagon in KG	12000KG
Maximum loading weight	The maximum total weight of a railway wagon plus loaded cargo	40000KG
Service interval	Determines the interval (time or distance) between maintenance.	37763km
Event-service (the last service status is based on the service event that has milestone end, with its date; repair history is given by collecting all service events of a wagon)		
Milestone	An indication whether a service is started or has been provided (end)	Start / end
Time	The time related to the milestone.	

Digital twin – wagon	The wagon for which the milestone is given	ID: RW0001
Digital Twin – maintenance service	Reference to details of the inspection (its stickers) and the type of service that has been provided (e.g. brake pads replaced)	WK0001
Event – train composition (the existence of such an event with a start milestone indicates a wagon is part of a train; the previous train can be derived by querying particular events in the past; the position of a wagon in a train can be derived by querying the position of its previous wagon)		
Milestone	An indication whether a wagon is part of a train	Start / end
Time	The time related to the milestone	
Digital twin – wagon	The wagon for which the milestone is given	ID: RW0002
Digital Twin – wagon or traction	The wagon or traction that pulls the previous wagon. In the example, the wagon RW0002 precedes the wagon RW0001 in a train.	ID: RW0001
Event – Wagon keeper (start and end indicate whether or not a wagon is still owned by a keeper)		
Milestone	An indication whether ownership relations exists (start) or ownership has changed (end)	Start / end
Time	The time related to the milestone	
Digital twin – wagon	The wagon for which the milestone is given	ID: RW0001
Digital Twin – organization/role	Ownership relationship of the wagon	WK0001
Event – service provision by Wagon keeper to RU (start and end indicate a wagon is still used by an RU, known as current RU)		
Milestone	An indication whether usage exists (start) or has ended	Start / end
Time	The time related to the milestone	
Digital twin – wagon	The wagon for which the milestone is given	ID: RW0001
Digital Twin – organization/role	The RU utilizing a wagon	TO67442
Event – cargo (the weight of the cargo and the empty weight of the wagon compose the current weight)		
Milestone	An indication whether a wagon is empty (end) or contains cargo (start)	Start / end
Time	The time related to the milestone	

Digital twin – wagon	The wagon for which the milestone is given	ID: RW0001
Digital twin – cargo	Reference to the goods or equipment (containers, trailers, etc.)	ID: cargoUUID
Event – inspection status (these can be queried over time)		
Milestone	An indication whether a wagon requires or has been inspected (start) or has repaired after inspection (end)	Start / end
Time	The time related to the milestone	
Digital twin – wagon	The wagon for which the milestone is given	ID: RW0001
Digital twin – inspection	Reference to inspection details. These contain the status of inspection and any remarks	ID: inspectUUID
Event – departure event (the time at which a wagon departed from a location)		
Milestone	An indication whether a wagon has departed	end
Time	The time related to the milestone	
Digital twin – wagon	The wagon for which the milestone is given	ID: RW0001
Digital twin – location	A physical location where the train has departed, e.g. marshalling yard	LocationUUID
Event – position (this can a location where the train/wagon has stopped; last and next stops/checkpoints can be derived)		
Milestone	An indication whether a wagon has passed a position or is at checkpoint	Start / end
Time	The time related to the milestone	
Digital twin – wagon	The wagon for which the milestone is given	ID: RW0001
Digital twin – location	A physical location which can be expressed in (for instance) GPS coordinates. In case more details of the location are available, a reference to that location is given (e.g. in case of a marshalling yard).	52.115391113240484, 5.289824337801916

The number of kilometers that a railway wagon has driven since its last service maintenance can be derived from data provided by an RU (via its TO) and the last service event that has taken place. For reasons of performance, it can be updated regularly by deriving it from events.

Train arrives at handover destination

RW0001 arrives at
Nedmag Veendam
Service Information

Part of TR20211201-17
Tractionair: TO67442

38527KG

Wagon status: OK
Next service: 37763KM
History: Brake Pads

The arrival event can be taken as an example. It basically refers to all relevant data, where this data is stored by a data holder. For instance, it refers to data of a wagon and via wagon to its cargo, its keeper, and the RU. Via the service – and inspection events it refers to the status and service history. All relevant data can be queried resulting in for instance the data shown here.

Any event is generated by one role and received by another. The arrival event for instance is generated by a Train Operator and received by an RU. The service data can only be retrieved by the RU when it has been shared by the Wagon Keeper that has received a link from its Wagon Maintainer provider. The actual weight of a wagon can only be given by combining the cargo data stored by the PLA with the weight of the wagon as stored by the Wagon Keeper.

3.4 Implementation by the BDI

When implementing the BDI, all data is shared as RDF or JSON(-LD). This implies that any other formats need to be transformed into one of these formats. A BDI node will have the capability to transform JSON(-LD) data into RDF.

Implementing the previous case with BDI implies:

- Each stakeholder requires to implement a BDI node for receiving and sharing events.
- Each stakeholder must make its data accessible via SPARQL. The following data holders are identified:
    - PLA – cargo details
    - RU – traction details, mileage of a train, and train composition
    - Wagon Keeper – wagon details
    - Wagon Maintainer – service details of a wagon
    - Infrastructure manager – checkpoints for handover
- The following events can be shared based on contractual relations (orders):
    - Position event – provided by an IM, shared with an RU, that can make it available to a shipper.
    - Departure event – provided by a Train Operator to an RU, shared with an IM, shipper, and PLA
    - Arrival event – provided by a Train Operator to an RU, shared with a shipper and PLD.
    - Service event – provided by a Service Maintainer to a Wagon Keeper, shared with an RU and potentially an IM.
    - Inspection event – provided by a Train Operator to an RU, shared with a Wagon Keeper and a Service Maintainer, potentially also with an IM.
- Search can be formulated on the individual BDI nodes by their owners. These searches will only give results if the BDI node owners have relevant event data and can access additional data.
- Identification, Authentication, and Authorization (IAA) needs to be implemented for data access by both a data user (generating an IAA token) and a data holder (verifying an IAA token).

3.5 Condition Based Maintenance Service

It is possible to develop a Condition Based Maintenance Service (CBM Service) for supporting various roles in rail transport for deciding whether maintenance is required for a wagon. Maintenance is currently scheduled on time intervals (time or number of kilometers since the last service interval); an algorithm (combined with sensors attached to a wagon) might improve maintenance.

Such a CBM Service can be provided to any role, for instance to a Rail Wagon Keeper that thus can improve its service to an RU, but also to a Train Operator for improving inspection. The CBM Service requires the implementation of a BDI node with links to data of various data holders, for instance (the list is probably not complete):

- Wagon details provided by a Rail Wagon Keeper.
- The last service event as shared between for instance a Rail Wagon Maintainer and Wagon Keeper.
- An accumulation of all kilometers that a wagon has driven as part of a train. This accumulation might have to be provided by various Train Operators that have used the wagon in a train since its last service event.

In case the service is offered to a Train Operator, the Wagon Keeper must enable access to paths driven by a wagon, where these paths are driven by other Train Operators. This can be done via agreements with the RUs using the wagons. Further research will be required to organize access to data.

A CBM Service collects data via sensors attached to railway wagons. These sensors provide a type of streaming data that serves as input to the service. The sensors can measure the number of kilometers driven by a wagon and collect data of the wagon status (brakes, etc.). The number of kilometers thus does not have to be provided by a Train Operator or RU, which decreases complexity of data access.

4 Steps needed towards a PoC implementation

This report tries to create an awareness of the application of blockchain technology as component of a BDI node to the railway sector. It is about wagon inspection and maintenance. It is a first step towards a more precise specification that needs to constitute of the following elements:

- Validating the FEDeRATED semantic model for sharing digital twins relevant to railway transport (e.g. wagon – and train details, but also locations like marshalling yards; these may already be implemented by the linked semantic model of the European Railway Association).
- Formulating the events that can take place for train operation (including wagon maintenance, path allocation, etc.).
- Formulating the business services that have been identified and providing examples.
- Linking particular events to business services, e.g. a service event is shared for a business service 'wagon maintenance'.
- Constructing a Service Registry by which all stakeholders can formulate their business services and provide an indication if they are able to generate the related events.
- Generating validation rules for the data that is shared based on the various events and relevant digitals twins.
- Formulating data requirements for a (data analytics) Condition Based Maintenance service and potential agreements to access relevant data.
- Developing rules for distribution of events and implementing these distribution rules in BDI nodes.

There will probably be more work to be done, this list requires further completion. It will be the basis for configuring BDI nodes for railway transport. A first technical experiment can be set up, simulating all roles in a process. Secondly, a business experiment can be made in a Living Lab setting, resulting in further exploration of on-boarding by users.

Finally, all railway specific parts that are required need to be included by the FEDeRATED semantic model and accepted by a relevant body like the ERA. This can (preferably) be done after the business experiment and before on-boarding new participants to ensure a solution that is stable.

There are already OEM (Original Equipment Manufacturers) of railway wagons that provide this type of service to their users, utilizing sensors attached to the wagons. These OEMs acts as railway wagon manufacturers also providing a maintenance service. They have all relevant data mentioned here and don't require the BDI. An independent CBM Service provider will require the BDI, since it does not have access to the data.

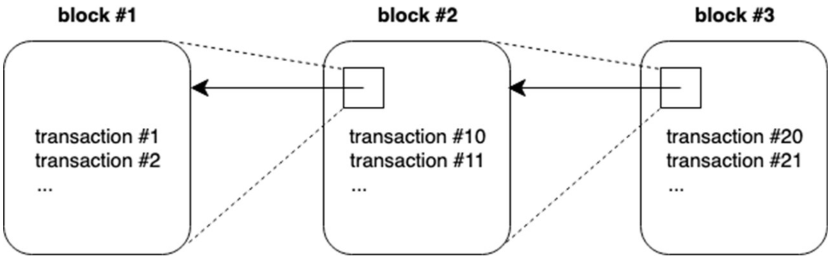
ANNEX I: Explanation of Blockchain

This is a brief introduction into blockchain. It is written for laypeople, people who have not studied computer science and typically regard IT as a black box. The style of this section is deliberately informal to lower the barrier of understanding.

A blockchain can be defined as a *decentral*, *irrefutable*, and *immutable* database of transactions.

Let's tackle this statement and start with a *database of transactions*. Back in the days companies kept track of their income and expenses by writing transactions down in a ledger. Once written down, a transaction is (almost) immutable, meaning that it cannot be removed or changed. Even today, a company's books should be closed after submission to the tax office.

Blockchain is a digital way to ensure exactly that. Taken literally[2], blockchain is a chain of blocks. Each of these blocks contains some transactions (much like a page of a ledger). In addition to transactions, each block contains a cryptographic hash, which acts like a fingerprint of the previous block. Alternatively, you can compare a hash to a picture of the previous page. This works like the famous Droste effect, the very last block will have a picture in which all previous blocks are visible. That ensures *immutability*, you cannot change a transaction without changing all the pictures following it.



The picture above visualises this mechanism of chaining blocks. If a block does contain a transaction that is not present in the picture on the last block, then you know someone made a fraudulent change. Sounds simple, but unfortunately, for a computer, it is a trivial task to change one transaction and update all the pictures in the following blocks.

To prevent such actions and create a digital ledger that is not only truly *immutable*, but also *irrefutable* (you should be able to prove that a transaction happened, also called *non-repudiation*) blockchains are *distributed*, or *decentral*. That means that there exist multiple copies of the same blockchain. These copies are called *nodes*. Each of the nodes can add transactions to the blockchain and they continuously share updates with each other through a *consensus algorithm*. There is much to explain about how exactly consensus is achieved, but for now it suffices to say that the *consensus algorithm* ensures that all nodes see the same blocks and the same transactions.

Blockchain enables the creation of a shared, transparent, immutable, verifiable, and irrefutable database of transactions, and that is a pretty difficult thing to do. If you work in logistics, then you probably have encountered situations where your data about a shipment differs from a supplier's

[2] We deliberately take a literal view for explanatory purposes. Blockchain technology, however, is often referred to as Distributed Ledger Technology (DLT), because not *all* DLT solutions use chains of blocks. All DLT solutions, however, do utilize some version of decentralized distributed data sharing with a consensus mechanism. In non-technical words this means that a DLT solution provides *proof* that something happened at a specific moment in time. The literal view of blockchain serves as a good explanation on how that proof is constructed.

data about the exact same shipment which can lead to disputes. A blockchain eliminates data discrepancy, and that is why it is sometimes referred to as a *single source of truth*.

Do note the paradox in that statement: To obtain an *irrefutable single source of truth* that truth should be *distributed* across multiple *nodes*, so in fact there are many sources of truth, but they all say the same thing.

With a standard centralised database (your bookkeeping software is likely to use one) this objective of *irrefutability* can be easily achieved if you trust the software and your accountant. The truth is backed up by trust. In a scenario with no central authority (like an accountant), where you share your database with other parties this is much harder to achieve. Blockchain technology was built precisely with the purpose of making it possible to achieve consistency, reliability, and irrefutability on a shared database in such a scenario.

We now understand what a blockchain is, but when should you use it?

Below are some good reasons to use a blockchain:

- Multiple stakeholders need to share and retrieve information in a scenario with no obvious central player:
    - *Do not use a blockchain if you are the only party sharing or the only party retrieving information, there are much less complex solutions for this. An example of this would be the sharing of container releases by a terminal. There are many parties retrieving information, but only the terminal is sharing. In this scenario the terminal is better off by building an API.*
    - *Do not use a blockchain if there is an obvious strong player who is also willing to take on a role of trust provider. In supply and logistics, there is no such strong player, there are many stakeholders; trust is implemented mostly based on framework contracts. There are of course authorities that can (should) be trusted, like ProRail*
- You have a need for immutable and timestamped information:
    - *Do not use a blockchain for information that does not need to be immutably timestamped. You don't need a blockchain to keep track of the addresses and contact details of your customers. Immutably timestamping this information could also violate GDPR regulation, as you need to be able to delete personal data.*
- You have a need for proof, for irrefutable information:
    - *Do not use a blockchain in environments where trust is not an issue, if you don't need irrefutable proof then you don't need a blockchain and you can rely on a normal database. There are (private) regulations like (e)CMR treaty that regulate responsibility and liability of stakeholders; these require data integrity that could be well provided by blockchain technology. In an open, dynamic organizational network (transactional relations), non-repudiation is a requirement, that can be met perfectly by blockchain technology.*

There are different decision instruments for selecting blockchain technology. We have also developed a way forward that can be applied (‘*The future of blockchain technology in supply and logistics*’, TNO, SPARK! deliverable, to be published).

ANNEX II: Use case development and deployment

Each use case like the one specified by this document, is based on a template. This template has been developed by SPARK! based on the various use cases (like the current one) and provided to the FEDeRATED project. The various use cases of SPARK! have also contribute to a more detailed specification of the components of the architecture, for instance event distribution in a case like the one for rail. This annex provides the latest version of this template and description of the BDI node; both will be updated based on new developments.

II.1 Expressing a use case in terms of the architecture

Of course, each use case requires a business case and has a lifecycle to come to full deployment (see '*Future of blockchain technology in supply and logistics,* TNO, SPARK! (to be published)). The guiding principles are leading in a use case; potentially a use case will not implement all of them. The latter depends for instance on on-boarding of new stakeholders, which requires the installation and deployment of a service registry.

In relation to a use case, the following information is required for configuring that use case in terms of FEDeRATED:

- Commercial relations – specify the commercial relations between stakeholders and visualize them by a transaction tree. This document provides an example. A transaction tree reflects the stakeholders (and their business services).
- Business services – identify the business services that are provided by individual stakeholders. Rail transport service is an example of a business service provided by a Railway Undertaking. These can be existing business service types already provided (e.g. business service type = transport) with restrictions to for instance the modality ('rail') and or ('equipment' = 'wagon'). See further.
- Business processes and event logic – there is a set of interaction types (events) that have a logical sequence and are of a type. These interaction types are grouped to business processes like 'booking' and 'ordering'. First, a selection needs to be made for each business service in the use case. For instance, commercial relations are based on framework contracts, which means selection of 'ordering' and 'visibility'. Next, three actions must be taken:
    - Select the applicable interaction types available via the provided choreography.
    - A combination of business service type (e.g. transport) and event type (e.g. order event) specify an interaction relevant to a use case.
    - In the context of a business service type and the mapping, the interaction sequence on an interface is defined. This is the basis for constructing examples of sequence diagrams. A (UML) sequence diagram is a standardized way of visualizing examples of interactions. It shows each role of a transaction tree by a line, where interactions can only take place between two commercial related roles in the transaction tree.

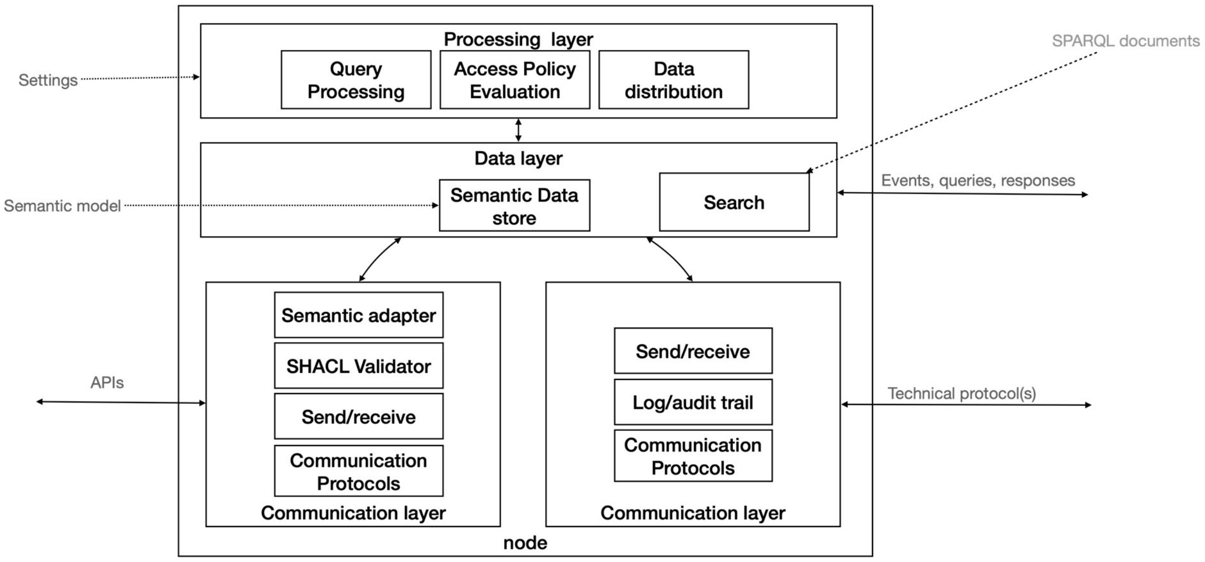
  It may be the case that the event logic of the use case is not yet completely supported by the choreography. In that case, the standard choreography might have to be adapted or a new one must be considered. The event logic is specific to a business service; constraints are specified by the minimal required data set of a business service and an event type.
- Semantic model. This about completeness and correctness of the semantic model to support a use case. The following aspects need to be considered:
    - Digital Twins – identify the subtypes of Digital Twins and analyse their completeness, e.g. goods, containers, wagons, vessels. Their completeness is based on the properties (are all relevant properties supported by the model) versus derived

concepts (like the kilometres driven by a wagon since its last service event), and any code values that are required.
    - Infrastructure – is the infrastructure fully supported by the semantic model (or aligned with the model like the railway infrastructure) and are all concepts/code values/identifiers available.
    - Events – a further specification of the event types (see event logic) in terms of the semantic model. These events may be the so-called atomic events or user events.
    - Roles/organizations – are all roles required for a use case part of the model and can all organizational details (and associations) be described by the model.
    - Linked data set(s) – specification of the data that can be retrieved when links are shared via events. This can be links to document data sets or data of a subtype of a Digital Twin. These linked data set(s) are formulated as SPARQL queries on the model.
- Regulations – if applicable, which country borders are passed by the physical chain, which modalities (Digital Twins) are involved, and what regulations are applicable in those countries. The latter requires details of the Digital Twins, like being waste or dangerous.
- Event distribution – a transaction tree with commercial relations is the basis for distributing events. For instance, a shared order event is the basis for sharing a position event. Additionally, a distribution mechanism for regulations might be required.

These choices result in configurations of components for a BDI node. This is described (briefly) hereafter.

II.2 Functionality of a BDI node

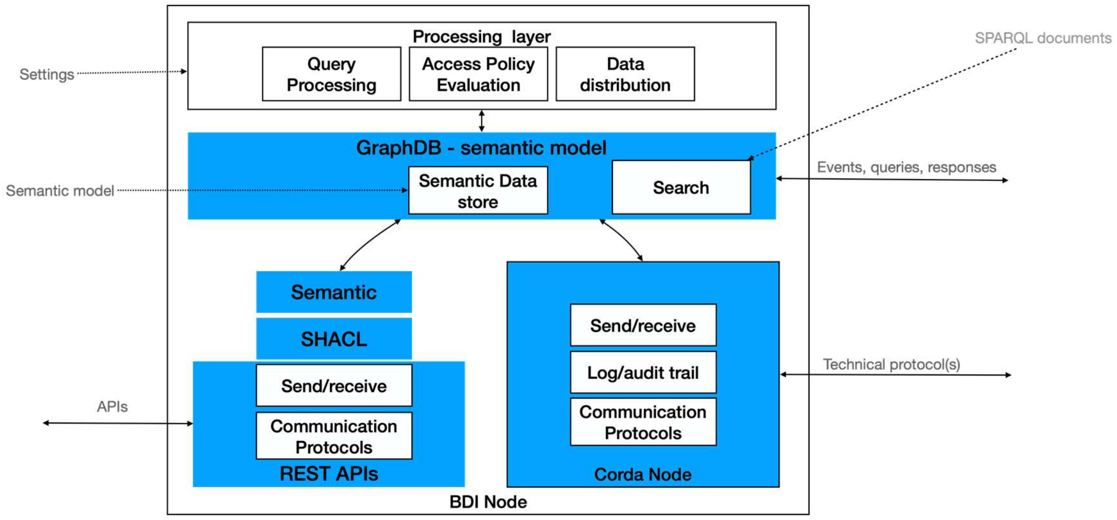
A BDI node implements the architectural components and supports peer-to-peer data sharing. The conceptual overview of functionality of a BDI node is given in the architecture document of FEDeRATED, it is depicted as follows:



(source: FEDeRATED Architecture)

The current version of the BDI node contains the communication layers with technical protocols provided by a Corda node. The Corda node also supports the log/audit trail via its notary network. The BDI node provides REST APIs to integrate with any internal IT system, supported by the semantic adapter and SHACL validator. The semantic adapter and SHACL validator can directly be configured from the semantic model, based on the linked event data sets and events. The semantic data store is implemented by GraphDB. GraphDB provides the Index functionality and contains the complete semantic model. SPARQL queries can be formulated on the semantic model.

The following figure shows the functionality of the version of the BDI node. The components in blue are currently supported.



Query processing, access policy evaluation, data distribution, and event logic need to be configured; they will probably be specific to a use case.

It is the objective to dockerize a BDI node (available 2022). A user that wants to become a user of the BDI needs to do the following (most complete for this moment; updates and changes can be made due to further development):

- Download the docker container with the BDI node.
- Install vm-ware for operating the docker container.
- Configure the internal security environment for allowing the BDI to connect with others and share data (this might be a firewall setting or can be more complex, depending on the internal IT environment).
- Install the necessary configurations for the semantic adapter, the SHACL validator and the semantic data store.
- Configure any specific searches on the semantic data store (Index) and construct a GUI or API for accessing the semantic data store.
- Configure the various REST APIs for integrating the BDI node with internal IT systems.
- Install the agreed query processing -, access policy evaluation -, data distribution rules, and event logic.

Additionally, the Service Registry is under development as it is specified by the FEDeRATED Architecture document.