UNIVERSITY OF TWENTE.

CGI

MASTER THESIS

By Arend Pool

Examination committee:

UT Dr.ir. H. Moonen
UT Dr. M. Daneva
CGI Ir. R. Voûte

October 2021

DIGITAL TWINS IN RAIL FREIGHT

The foundations of a future innovation

Acknowledgements

It is 12 AM, I am sitting at the bar table in my new apartment, music fills the living room and the dimmed lights make for a cosy workplace. It is the setting that depicts the ending of a project that has occupied me for more than 6 months straight: an adventure that has made me explore all emotions on the spectrum. I started this challenge knowing I would enter a field unknown to me with the motivation that I got to work out an innovation that excites me. I have learned so much during the research, I got to talk to very knowledgeable people and I am proud of the work that marks the end of me as a student.

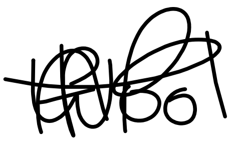
The past few months I have been tested on many fronts: my capability of understanding the business side of processes, my capability of understanding and designing IT environments, my ability to think and argue academically, my endurance and resilience and more. Finishing this work strongly is of importance to me as it gilds the journey I have had at the University of Twente.

Researching the fields of rail freight and digital twins through conducting design science have taught me more than just plain knowledge. Of course, I have learned that the IT landscape is very complex in the chain, or that digital twins are way more than visualisation, but the real valuable knowledge is more indirect: I have learned how to judge information more scientifically, how to explain intricate concepts more clearly and most of all I have learned many new characteristics of myself.

I want to take a few words to thank three highly involved supervisors, to whom I completely owe my gratitude. Without Hans Moonen (University of Twente), Maya Daneva (University of Twente) and Robert Voûte (CGI the Netherlands) this piece of work would have not been the success I was aiming for. Their sharp feedback, motivational talks and understanding attitudes got me back on track in times that I had lost willpower. Furthermore, I want to thank Sander Kapsenberg (CGI the Netherlands) for all the effort in thinking along and providing me with the very much needed contact network. A thank you is in its place to Arjan Vonk and Mels Smit (co-students) for the useful feedback meetings from the very start to the very end. I want to thank my sweet girlfriend Nienke for motivating me lovingly throughout the extent of the project. And finally, a very deserved thank you to my parents who supported me unconditionally through the extend of my entire study and give me the resources to pave my way.

To you, the reader, whoever you are, thank you for taking the time to read my work. I hope there are plenty of insights or guiding references that could help you in any way. Feel free to use any information or model that is useful, just make sure to give the credits where due :).

I wish you a happy reading, all the best from me, the author,



Arend Pool

Abstract

Environmental concerns drive the shift to rail transport over trucking transport which is putting pressure on the European rail network. While goals of doubling the transported mass on a stagnant infrastructure may sound very ambitious, these *are* the goals set to be accomplished by 2040 in the Netherlands. As doubling the total rail length is no viable option, there is a strong demand to seek smarter ways to increase capacities on the current network.

The present master thesis addresses the demand for creative solutions towards increasing rail network capacities. It presents extensive research aimed at extending current knowledge by contributing a solution grounded on digital twin concepts. The thesis provides a well-founded answer to the research question of "*what constitutes a good digital twin design to solve transparency-related issues in Dutch rail freight?"* Specifically, this research proposes a design for a digital twin for transparency-related issues, intending to increase capacity on the current network. Following a multitude of research methods, this research triangulates the specific problems of the current IT landscape that could be solved by designing an architecture for a digital twin. Using literature reviews techniques and extensive case study research in the Dutch rail freight sector, a minimum-viable-product design for a twin is proposed. The development and the empirical evaluation of this proposal implemented the guidelines for Design Science Research as per Wieringa (2014). Validity aspects of the proposal were investigated through two expert panels.

The design science process includes three stages: problem analysis, solution design and solution evaluation. These are summarized as follows:

First, by modelling out chain-wide processes, landscaping the IT architecture and reviewing complaints from within the field, the most prominent problems were determined. Transparency is found to be the main theme, but not only with the focus on capacity management. Environmental concerns are also existent in the form of the transport of hazardous materials, for which caution is needed to divert unnecessary risks with potentially catastrophic consequences.

Second, the solution design stage. Capacity- and incident management were chosen to be the guiding perspectives for solving the design problem that drove the design of the newly proposed solution architecture. A six-layered construction is the outcome of the design phase, grounded on a validated scientific model with the needed extensions for increased modularity, a term that has found its significance in such an innovation. Furthermore, due to cultural aspects and costly developments, a rollout of such a system should come gradually. For this reason standardisation, centralisation and the model-view-controller principle have been incorporated in the solution proposal. In turn, the resulting architecture relies on an "event"-based mechanism, in which events will be immutably stored in chronological order and processed into a relational database scheme.

Third, the validation stage. It was found that design-level requirements came too early at this stage, and the focus should first be on the first 5 layers of the architecture, rather than on the visual sixth layer. While the higher-level designs showed promise and seemed to have similarities with examples of practice, one addition had to be made: it was concluded that many third-party applications (such as the ones of the network operator) also make use of an event-driven architecture that should be incorporated in the event hub, the additional layer to the architecture that showed to be indispensable.

This thesis has some important implications for practice and research. An important finding in this work is the focus that the architecture should get: while at the beginning of our research process, it was hypothesized that the "View" layer would be the most important as it is the direct bridge between raw data and cognitive knowledge, it later was found that this level of design was way too low at this point and that it is the

other five layers that deliver the substantial value for this stage of research. Due to the higher-level design scope as set in the conclusion chapter, the research could be scaled up fairly effortlessly in upcoming research. This also makes the system more viable: the high-level design is ready to get implemented in a test setup to be validated empirically. Due to the “lean” approach design of the modular system, scaling up from these simple test setups would be uncomplicated.

This thesis lays out the foundations of research in operational digital twins in rail freight, which extends the current knowledge that regards mostly asset-focussed digital twins. This is done by providing the research community with blueprints of operations and applications that compose the rail freight chain and using these blueprints to identify unsolved challenges in the field. Among the challenges, the most outstanding one – the matter of transparency – has been used to design the digital twin, showing how a twin could be a solution to transparency-related issues in rail freight. The next steps in research and practice would be to create a low fidelity prototype to serve as a proof of concept. This could evolve gradually into useful proof of concepts with elicited design-level requirements.

Contents

List of figures

List of tables

1 Introduction

An ever-increasing demand for transportation has transformed logistics into a never-ending flow of goods on multimodal networks in complex chains. Our lives have become more luxurious and materialistic, while environmental concerns arise from our acquisitiveness. We have to apply greener parameters to our transport route calculation, as also compelled by the EU: pressure is put on the logistics branch to shift from the roads to the rail- and waterways. This is where new problems arise, as the current rail infrastructure risks getting congested which makes for inefficient transport modes. Process optimisation is needed, where IT solutions provide insights needed to utilize every inch of the network.

Consultants at CGI the Netherlands, a consultancy firm with a great footprint with large Dutch railway organizations, have philosophized about having a Digital Twin to provide clarity on what happens on the rails (Moonen H. , 2021). Information Technology is unthinkable in the branch, and the widespread of information technology solutions form an intricate network of interconnected applications. Could a centralized view be useful for coping with the increasing traffic on the railways?

In this research, it will be assessed how such a design of a digital twin should look like for these transparency related issues. First, a thorough understanding of the context and the problems will be generated by modelling the business and the IT in the logistics chain: processes will be mapped onto the supporting applications. Then, the bottlenecks in the IT landscape will be assessed for which a digital twin could be a solution. This will be the foundation on which a design will be made, validated, revised and proposed.

1.1 Research context

This research is proposed by CGI, an originally Canadian consulting firm with 400 branches in 30 countries. The specific firm that has requested the assessment of digital twins in rail freight is located in the Netherlands. The Dutch branch has many large active projects in Dutch train traffic, both on passenger traffic and freight traffic. The Dutch Network Operator is an important customer to the firm, and many projects on the IT level are performed by CGI Netherlands. This is also where the case study will take place: interviews will all be held in the scope of the Dutch rail freight chain.

Dutch rail freight is a new field to the researcher, which is why the research path taken is quite broad: many different aspects will be assessed. Rail freight is international, which is why processes in the continent are similar: standardisation is pushed by the EU. The scope of the research will therefore exceed the Dutch borders.

The year of writing is 2021, and the term “Digital twin” is getting somewhat more cynical names such as “buzzword”. Taking the hype cycle as a perspective, this is expected, as in 2018 the “digital twin” concept was said to be at the peak of inflated expectations (Eyre & Freeman, 2018). Even though digital twins are promising, it must be noted that the concept is quite freely interpretable, and is not necessarily the holy grail. Because the concept is quite broad, first some assessments on the different types of twins will be provided (Section 2.3).

1.2 Problem statement

The extended complexity of logistic chains have made the alignment of assets, people and processes increasingly difficult. Intermodality has exposed the field to new challenges regarding operational alignment including many different parties. Transport networks have become congested as the mass of transported goods increases with a multitude compared to the growth of networks (Cambridge Systematics, Inc., 2007). It could be said with certainty that the network cannot keep up with these growths: the ambitious goals of governments with regards to surging the used capacity on rails are impossible to achieve through extending the network.

Even though innovations arise to support these developments, such as tracking and tracing of containers (Kia, Shayan, & Ghotb, 2020), RFID identification (Rosova, Balog, & Šimeková, 2013) or a centralized application of port-wide information systems[1], the rail freight sector is still conservative (Wiegmans, Hekkert, & Langstraat, 2007). Transparency of the rail freight chain is lacking (Moonen H. , Gevaarlijke stoffen juist op het spoor, 2018), even though research has been conducted on the tracking and tracing of trains and wagons (Ulianov, Hyde, & Shaltout, 2016). Ulianov et al. (2016) have worked out complete tracking and tracing models, but the fact that there is a political side to data sharing has not been incorporated. Little is known about transparency and centralized system integration in rail freight, while also incorporating the willingness of parties to share data.

Centralized system integration is an issue that needs some special attention. Due to logistics being international with many involved actors, and with each of these actors administrating their applications, integration and centralization of IT systems is a challenge that has to be overcome. For centralization, one should know about the core IT infrastructure that is to be integrated. A paper addressing all software and interconnected business operations in rail freight has not been found in the literature.

This transparency issue seems to be of real concern with respect to the transport of hazardous materials (Moonen H. , 2018). Despite the goals of increasing rail capacity coming from environmental concerns, increased rail capacity also provides new environmental concerns. Hazardous materials on rails have been a matter of interest to researchers: a routing strategy has been developed to level out costs and risks (Glickman, Erkut, & Zschockec, 2007), a model to estimate risk reduction for several strategies have been designed (Liu, Saat, & Barkan, 2013) and more strategic planning and routing methodologies for the transport of hazardous materials have been defined (Ke, 2020). These researches have been on risk assessment or prevention in the pre-operative phase. Not much research has been done on the assessment of risks in the operations itself.

Digital Twins are the propagators of transparency and centralized integration. Even though sometimes perceived as a “buzzword” (Surianarayanan, 2020), the many use cases in different industries have shown the potential of digital twins, and there is no doubt that the concept will keep getting its deserved attention. Where digital twins have been used in several industries for tracking and tracing objects in a system, the research conducted regarding digital twins in rail freight has mostly been on asset-health monitoring and predictive maintenance (Anylogic, 2020) (Surianarayanan, 2020). Perhaps digital twins could also provide a solution to the issues related to transparency in rail freight?

1.3 Scope

The research of the given problem statement has some clear boundaries that need to be defined to maintain a research focus and to allow generalization. The latter is achieved by making sure related work falls in the same scope, and therefore the scope should not be too broad. Generalizing the outcomes of these objects should be done in the context of all rail freight chains that incorporate the stakeholders that are involved in the application landscape model (in Section 5.2).

[1] https://www.portbase.com/en/

The given solution that is the outcome of the main research objective will apply to all parties part of such a logistics chain (as given in Section 7.1) that need to find a solution to transparency related issues. This objective will define new limits to the research by providing clear unambiguous problems to be solved by designing the system. The circumscribed scope will be given in the form of a Wieringa (2014) design problem in Section 6.2.

While the assumption is that many rail logistic chains operate similar to the use cases given in this thesis, given the cross-border nature of the field, it should be mentioned that such a twin would not necessarily provide all of the given benefits to all rail freight contexts. This is why the limitations have been set to the context of alike chains, in other words, having the same involved stakeholders with the same responsibilities.

1.4 Research questions

To get straight to the point, the main research question of the matter is stated as follows:

What constitutes a good digital twin design to solve transparency related issues in Dutch rail freight?

1.4.1 Sub questions

This section takes a top-down approach. The research background as described in Chapter 2 of the thesis is at the broadest scope and merely acts as the basic knowledge to understand the more in-depth research. Narrowing the scope is the goal of the theoretical research: after defining the problem context and the stakeholder context we can triangulate the most substantial problems to craft a design problem (Chapter 6). The design problem in the given template (page 28) should be deduced from the answer to one of the research questions, therefore we could state the question as follows:

- How could digital twins be a solution to information inadequacies in rail freight? (SQ4)

Some information deficiencies are found in this question, aspects that need deeper research. These information inadequacies with an emphasis on digitalisation (after all Digital Twins are a digital solution) have to be figured out. This needs an understanding of the extent of digitalisation in the chain and of course the underlying chain mechanisms. To answer the aforementioned sub-question we need to know:

- Where is digitalization lacking concerning information needs? (SQ3)

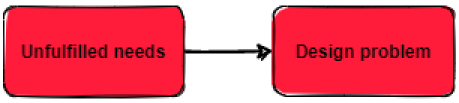


Figure 1.1: Unfulfilled needs in relation to design problem

The relationship between the questions is shown in Figure 1.1. Still, there is a need for more in-depth research. First, it should be researched how the information flows through the chain. This gives both an understanding of how the logistics chain mechanisms function, and a reference frame of where the inadequacies may lie. Furthermore, it will provide an overview of related stakeholders. Although, even if the communication flows are known, how to determine and scientifically certify any mentioned inadequacies? Knowledge about the extent of digitalisation is still unknown: what parties administer applications? what applications are (not) supporting the communication flows? And what are existing innovations? The final two sub-questions to complete our theoretical framework are:

- What communication flows characterize the rail logistics chain? (SQ1)
- What IT architecture supports the business needs in rail freight? (SQ2)

SQ1 will provide some of the necessary insights that will be applied to SQ2 (such as the stakeholder descriptions and the alignment of interrelated processes. The relationships between the questions in this investigative phase are shown in Figure 1.2.

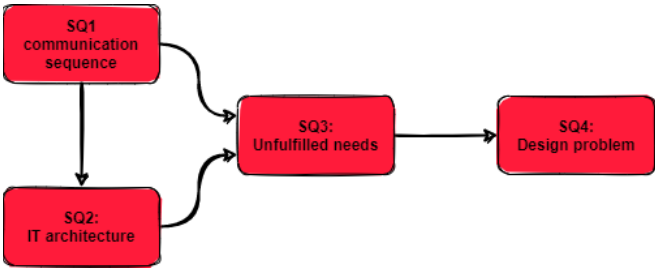


Figure 1.2: Relationship between questions in the problem investigation phase

Once the design problem is construed, the research has a solid base to build the artifact upon. There is a clear distinction to be made here: the treatment to be designed and the treatment to be implemented. The goal of the research is to check whether a certain design could serve as a treatment to the design problem, but it is the verification of the treatment that answers the topic in question. Therefore, this research will define the artifact as an MVP that would suffice when fully implemented, but for validating the treatment only a demo would be designed. The extent to which the concept should be implemented for validation is dependent on a few issues: what the arisen problems to be solved imply, how current operations are performed and how these operations could be improved by the artifact. Based on that, the empirical cycle could be implemented through research design, but this will take some further investigation first. The question that will lead to an answer to this phase of the engineering cycle is:

- What design choices characterize the digital twin? (SQ6)

The relationships that follow with the are shown in Figure 1.3.

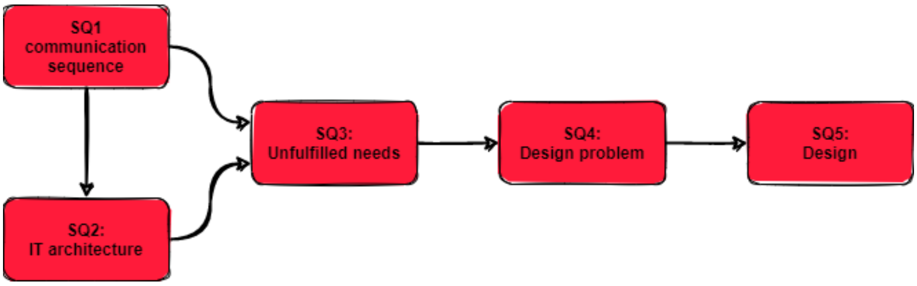


Figure 1.3: Relationships between the questions in design phase

Validation should take place in the early engineering stages; the goal of this research is to scientifically propose minimum design choices as a solution to encountered problems in the scope of rail freight digitisation. Even though we do specify the artifact in detail, we recognise the need for scientifically grounding the design in an upscaling fashion. The questions that will provide the focus of the validation phase are basic:

- Is the designed digital twin design an adequate and useful solution to the problem? (SQ6)

All of the sub-questions (1-7) are displayed in Figure 1.4.

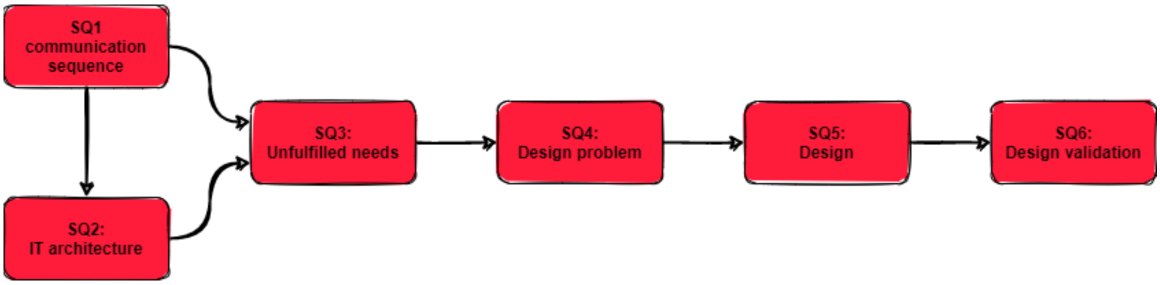


Figure 1.4: Relationships between the research questions

1.5 Objectives

The primary objective of the research will be:

Setting out a basis for future works on digital twins in rail freight by defining the minimum system requirements.

Designing such a system aims to integrate several systems, which calls for a need for an overview of systems and processes along the entire chain. These processes first have to be determined and mapped, as no clear overview of processes during the transport on railways is provided by researchers yet. This is where the first two sub-objectives lie of this research, of which the contributions would be schematic models of how actors, processes and applications are correlated in the entire chain:

1. Creating a systematic and chronological diagram of operations in rail freight, and
2. Generating a cross-chain blueprint of the IT infrastructure in rail freight.

Then the research will point out where the specific challenges lie in rail freight regarding the IT landscape. These challenges, or shortcomings for that matter, could be used by other researchers to extend the design of the digital twin or to get inspired by for designing new solutions. The next sub-objective is:

3. Providing the research field with challenges in rail freight on IT level that need more scientific attention to be overcome

And finally, besides delivering a design for a digital twin itself, the designs come with their contributions. They show how digital twins could be solutions to transparency related issues, one of which is the monitoring of transport of hazardous materials. These final two contributions could be formulated as:

4. Showing how transparency could improve risk assessments in operations, and
5. Showing how a digital twin could be a solution to transparency issues in rail freight, while also incorporating the governing challenges such as willingness to share data and sensitive information.

In Table 1.1, an overview is given of the structure of this report: it shows which objectives are achieved through answering what specific sub-question, and the chapter in which the matter is discussed.

Phase	Question	Chapter	Objective
Problem investigation	SQ1	4	1
	SQ2	5	2
	SQ3	6	3
	SQ4	6	4
			5
Treatment design	SQ5	7	Primary objective
Treatment validation	SQ6	8	

Table 1.1: Overview of questions, chapters and objectives

2 Background

In the "Background" chapter the reader is presented with the necessary theoretical knowledge base to understand the research that is being conducted. The chapter first gives an abstract image of what rail freight looks like: economically, environmentally and operationally. Then the overall concept of digital twins is illustrated by showing different applications and benefits, and by discussing a scientific framework for designing a digital twin's architecture.

2.1 Rail freight

Logistics is about managing never-ending flows of goods in a chain of aligned processes. These goods have destinations and origins, goods have owners, and goods have value to these owners. These and many other parameters determine how certain goods ought to be transported as there is not a singular mode of transport but several different modalities are found. This section describes how rail traffic is mostly concerned with the hinterland transport of heavy bulk and hazardous materials. Furthermore, the section shows how rail logistics should be more efficient to solve environmental issues.

2.1.1 Market share

According to Centraal Bureau voor de Statistiek (CBS), the Dutch national bureau of statistics, train freight is responsible for just 3.4 per cent of the total transport mass of the Netherlands. In comparison with other European member states, the share of rail transport in the Netherlands is limited but is highly compensated by the large share that barge shipment offers (Eurostat, 2018). Even though rail transport is the greener solution, barge transport is still way more energy efficient when compared to road and air shipment (Responsible Care, ECTA, Cefic, 2011).

In the statistics of CBS four modalities were incorporated: road, barge, air and rails. The modality with economically the largest impact would be the sea modality, but this will not be included as the sea modality is not a competitive modality to rail freight. Road, barge, rail and in some cases air or pipelines are so-called "hinterland modalities", as they ship goods from the ports where global transport is inbound, to the hinterlands.

The market shares in both mass and value as provided by CBS are shown in Table 2.1. According to CBS, trucks are responsible for about 82 per cent of the domestic freight and barge is responsible for at least 17 per cent. This does not leave much room for rail transport, which only has a share of half of a per cent of the total transport mass.

Modality	**International freight mass (%)**	**Domestic freight mass (%)**	**Total freight mass (%)**	**Total freight value (%)**
Road	46.1	82.2	68.3	74.5
Barge	45.4	17.4	28.2	9.7
Rail	8.1	0.4	3.4	5.2
Air	0.5	0.0	0.2	10.7

Table 2.1: Freight masses and value compared per modality in the Netherlands (2019)

CBS does show an upward trend of the total mass of transported goods on rails, but this cannot be directly related to a modal shift: the trend of transported rail freight is very similar to the trend of the total transported mass that is displayed in Figure 2.3. In terms of international freight, a more significant presence of the share of rail freight is uncovered, but could still be regarded fairly low compared to barge and road transport. Where the share of rail is almost 8 per cent higher than with domestic freight, the share of road and barge has a more balanced distribution.

2.1.2 Role of rail freight

Air transport only has a low share of the total transported mass but is very relevant nonetheless: even though the total transported mass on rails is larger, the total cargo shipped by air is double its value. Especially when compared to barge transport the difference is astonishing: it implies that cargo of high value is shipped through the air and that heavy, low-value cargo is shipped by the relatively cheap barge modality. This was also found by Otten (2020), who found that rail freight is most responsible for the transport of coal, iron, organic chemicals and plastics. The high value of air freight could be due to the service provided by the modality: agile, reliable and safe (Meng, 2010). The additional service provided by air transport is worth its value for more costly cargo.

Sea carriage is aimed at long-distance international transport and could become a long-term competitor for rail freight due to the increasing developments of train connections with China. Even though many challenges are ahead, such as increased theft risk and the lack of transparency, the China corridor is developing (Islam, 2013). For the sake of rail transport, this could be a positive thing, for the sake of the Dutch economy, it is questionable. In the “goederenvervoer agenda” (Ministerie van Infrastructuur en Waterstaat, 2019) this is mentioned as one of the drivers for development to keep the Port of Rotterdam attractive for Mondial transport.

The role of trucking transport in the logistics chain seems to be quite distinct: domestic freight. Even though barge plays an important role in transport, both domestic and international, road shipment is very dominant in the domestic sector. This is due to the freedom that the roads offer, the 140.000 kilometres of the road provide way more efficient and accurate transportation than the 6.000 kilometres of waterways, let alone the 3200 kilometres of railroads that are accessible for freight (Centraal Bureau voor de Statistiek, 2019). This freedom makes the modality of perfect use for door-to-door transport, whereas rail and barge have better use in intermodal transport. Considering barge shipment, we see its share of both transportation scopes is nearly equal.

Trains are more of use in international shipping of heavy cargo and the transport of hazardous goods. According to the central government, 10 per cent of all train freight in the Netherlands could be classified as dangerous or polluting (Rijksoverheid, 2019). Dangerous goods, often in the form of liquid bulk, have an impact on logistics in the form of regulations. For instance, dangerous substances are banned from the emplacement Waalhaven Zuid due to failed safety requirements (Verwater, 2020). Many regulations have been put on the transport of hazardous substances, through the Dutch laws and regulations transport hazardous substances and the European guidelines concerning land transport of hazardous substances.

Figure 2.1 provides an overview of all the import and export flows in the Netherlands: the infographic shows the role and share of each modality concerning international shipments, and also puts the significance of the port of Rotterdam in perspective. A clear distinction can be seen between rail and road transport: rail is used more for outgoing services, whereas trucks are the most used for incoming transport. Goods from Rotterdam are likely destined for the hinterland and the empty wagons are shipped back to get new goods from Rotterdam. This could be where opportunities exist for the future of rail transport by transferring incoming shipments from roads to rails.

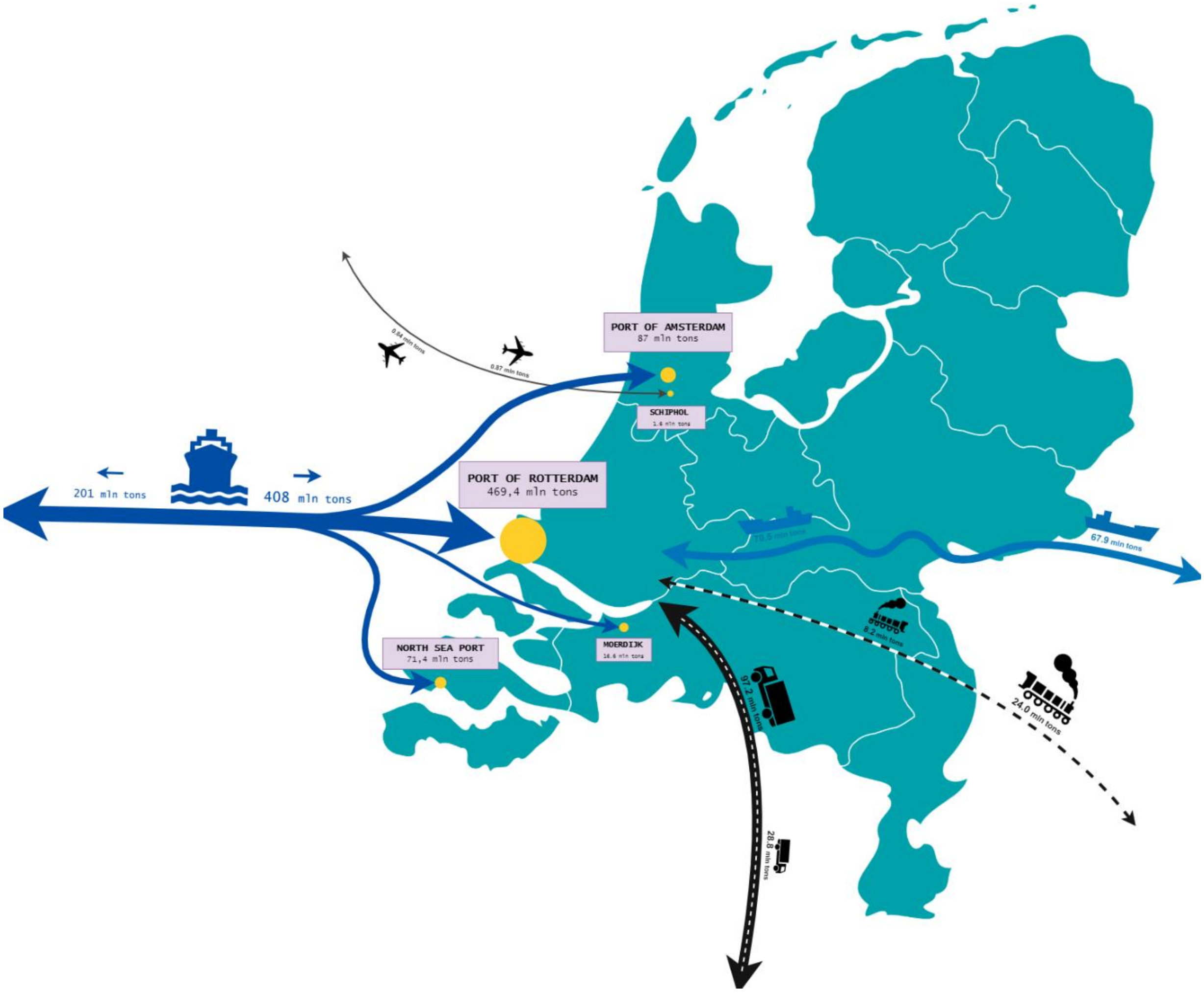


Figure 2.1: Freight flows in the Netherlands (Data from CBS, as of 2019)

2.1.3 The logistics agenda

The Mondial role of the Dutch logistics system is quite significant and established, but is not necessarily secured and future-proof according to the Dutch Ministry of Infrastructure and Water Management (2019). Pressure is being put onto the rail network and pollution awareness is rising as both freight and passenger transport grow (Centraal Bureau voor de Statistiek, 2019). Political agendas have incorporated the need for more efficient ways of transport and are trying to stimulate development in a greener direction (Ministerie van Infrastructuur en Waterstaat, 2019) (Ministerie van Infrastructuur en Waterstaat, 2018) (Directorate-General for Mobility and Transport, 2019). Figure 2.2 shows the CO2 emissions per modality and Figure 2.3 shows the upwards trend of the total transport mass in the Netherlands. We could say that, at least in the Netherlands, transport has become more “green”, however, to reach the ambitious zero-emission goals set by the Dutch ministry and the EU transport has to become even more efficient.

The EU plead for two key subjects on which most of the investments should be aimed (Directorate-General for Mobility and Transport, 2019): the completion of the European rail network and investments in multimodality transportation methods. With investments in the network also investments in the IT solutions are incorporated. The fact that the EU aims to invest in multimodality implies a more thorough use of the maritime and rail infrastructures. Trucks are inefficient in terms of emissions as they are responsible for almost three-quarters of the entire transport pollution of the EU and are causing major bottlenecks on the road network. The EU has recognised this problem and initiated the Marco Polo program to focus on the shift from the road to other modalities through policies. This program aims to reach the goal of zero-emission hinterland freight (Otten, 2020). The potential environmental benefits are quantifiable by

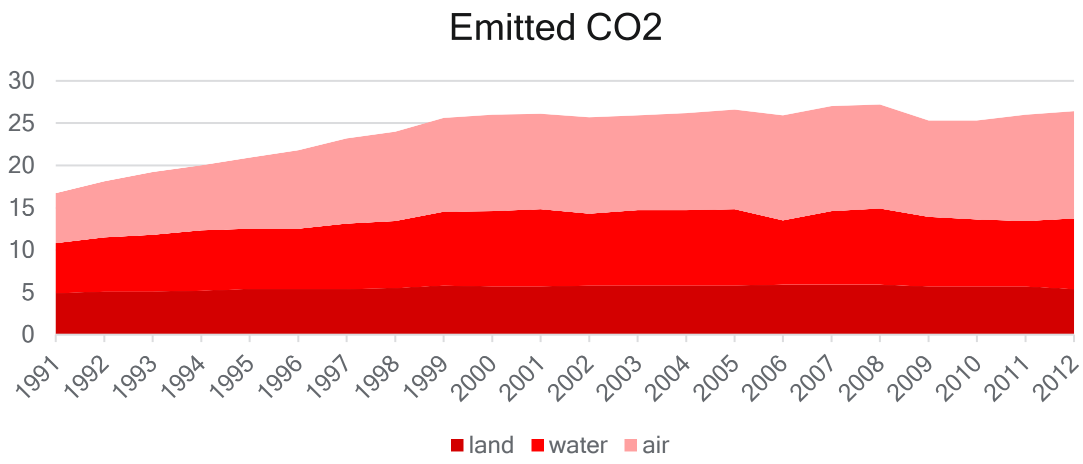


Figure 2.2: Total emitted CO2 per year per modality

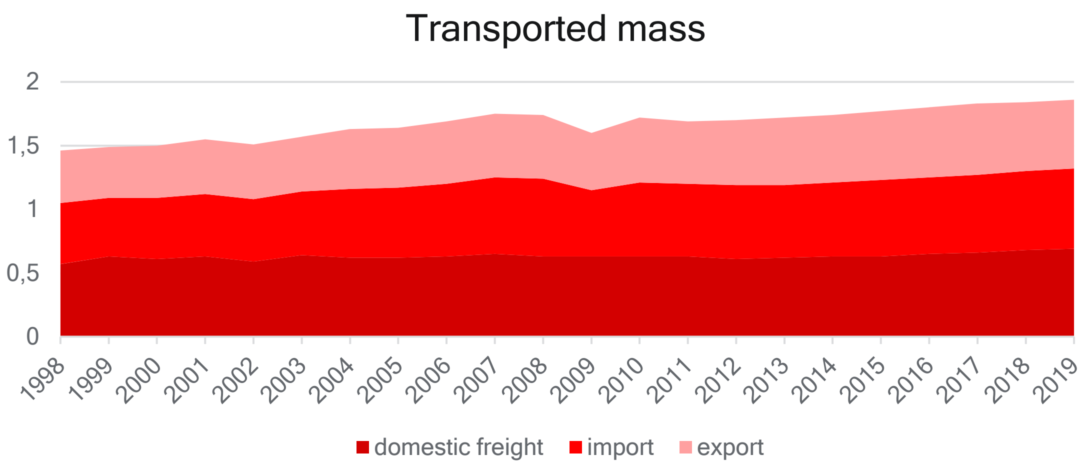


Figure 2.3: Total transported mass per year

using calculations reported by (Responsible Care, ECTA, Cefic, 2011), which identified methods for calculating transport emissions. According to their findings, road transport has an estimated average pollution of 62 grams CO2 per tonne-kilometre. To put the potential in perspective: the estimates for barge transport are 31 grams CO2 per tonne-kilometre, for rail transport 22 grams CO2 per tonne-kilometre and sea transport only 5 grams CO2 per tonne-kilometre.

Not just the EU has recognized the upside potential of intermodality; the Dutch Ministry of Infrastructure and Water management also highlights the importance of switching to modes other than truck shipment. They have set ambitious goals of almost doubling the freight moved on rails by 2040 (Ministerie van Infrastructuur en Waterstaat, 2019). Because the growth in rail infrastructure is very limited due to the population density in the Netherlands, this has quite a significant impact on the entire rail sector. Not being able to double the rail capacity implies the need for more efficient use of the current infrastructure. One way that this could be achieved is by increasing train size. In “Goederenvervoeragenda” (2019) the Dutch governments plead for the use of trains up to 740 meters long.

Upscaling rail capacity shows the importance of IT solutions for rail management to avoid congestion in the network and to ensure all chain parties operate smoothly, which is exactly what another high priority theme of transport agenda entails (Ministerie van Infrastructuur en Waterstaat, 2019): digitalisation and automation of freight transport and logistics. The ministry has set goals to maximize efficiency through digital solutions, such as harmonized data exchange, data sharing among parties through platforms and paperless transport. This allows for more dynamic and flexible train services and makes reactivity to unforeseen events more efficient, allowing for more capacity on the rails. A Dutch research institute pleads for the importance of digital innovation in the rail industry: according to them, innovation in the road transport sector are going way faster than in the other modalities, and therefore a reverse modal shift (from rail to the road) is a real danger (TNO, 2018).

2.2 Cargo journey

Logistics is more than getting your goods from one place to the other, nowadays it aims to have a moving inventory, as said in an interview with a large international rail Carrier (Appendix E). In this interview, it was emphasized that an important distinction has to be made when comparing train freight with passenger trains: in logistics, there are mostly fixed flows of cargo. It is a harmonization of goods propagating through a complex network of which rail haulage is only a part of the chain. This section shows how the concepts of intermodality and modality shifts work schematically (Figure 2.4), and ties these concepts to the journey that cargo travels through the chain. The section gives a short introduction to the operational processes that are relevant to the thesis, which will be assessed in detail in a later stadium (Chapter 4).

> *"There is a very big difference that you have to understand. With normal passenger transport, you are an occasional traveller. We barely not know this flow in freight traffic, you always have a contract there." – Rail Carrier*

2.2.1 Inter- and intramodality

Xie (2009) call the development of the transport networks complicated and multidimensional: the networks have gone from parallel, small ports, to the use of inland ports towards an interconnected network of intermodal transportation systems (Xie & Levinson, 2009). The use of inland ports is referred to as dry-ports by Roso (2009), who capture the different types of inland ports by showing clear abstract figures of the networks. The dry port concept as described by Roso (2009) is schematically displayed in Figure 2.4, where inter- and intramodal transportation modes are displayed. Node C in the figure is an example of an inland dry port. Furthermore, dashed lines represent intramodal modes, where, for example, trucking companies deliver to one another. Intramodal terminals in rail logistics are emplacements (D in the figure), in which wagons

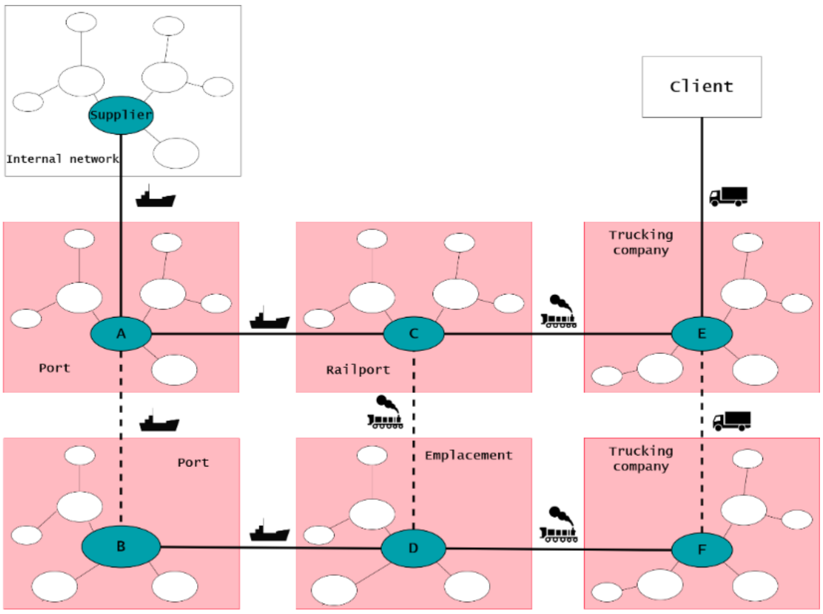


Figure 2.4: Intermodal transport network

are shunted on a complex network of railways and attached to locomotives travelling to their corresponding destinations. The intermodal terminals are junctions in which transhipment occurs: the transfer of goods from one modality to another. The Rotterdam port is an example of this intermodal terminal: sea freight gets transhipped onto trains, barge ships, trucks or pipelines. Transhipment of the goods is done within the internal network of each node.

2.2.2 Operations

Rail haulage includes several types of operations performed at different types of locations. First of there are inland terminals, which allow solely for the turnover of container freight. Then we have rail ports, which also allow for the transhipment of bulk goods (Kennisinstituut voor Mobiliteitsbeleid, 2019). And finally, we have emplacements, where shunting is performed. The journey of the transportation of goods is depicted in Figure 2.5, which has examples of ports, terminals and emplacements. When goods are transported for a Client, the cargo gets hauled from a supplier to a port or terminal, depending on the type of freight. The train Carrier dedicates a wagon of whatever type is needed to the Client, after which the shipper inspects the wagon and loads it (Fernández L., 2004). Freight might arrive at the rail port through any modality, where it gets transhipped. Vis (2003) describe the operation performed regarding container transhipment at ports thoroughly. First, when goods arrive, they are piled up and stacked before being transported to the transhipment machines. This is called drayage, and is responsible for a large fraction of the transportation expenses, at least regarding rail line haulage (Bontekoning, 2004) (Fernández L., 2004).

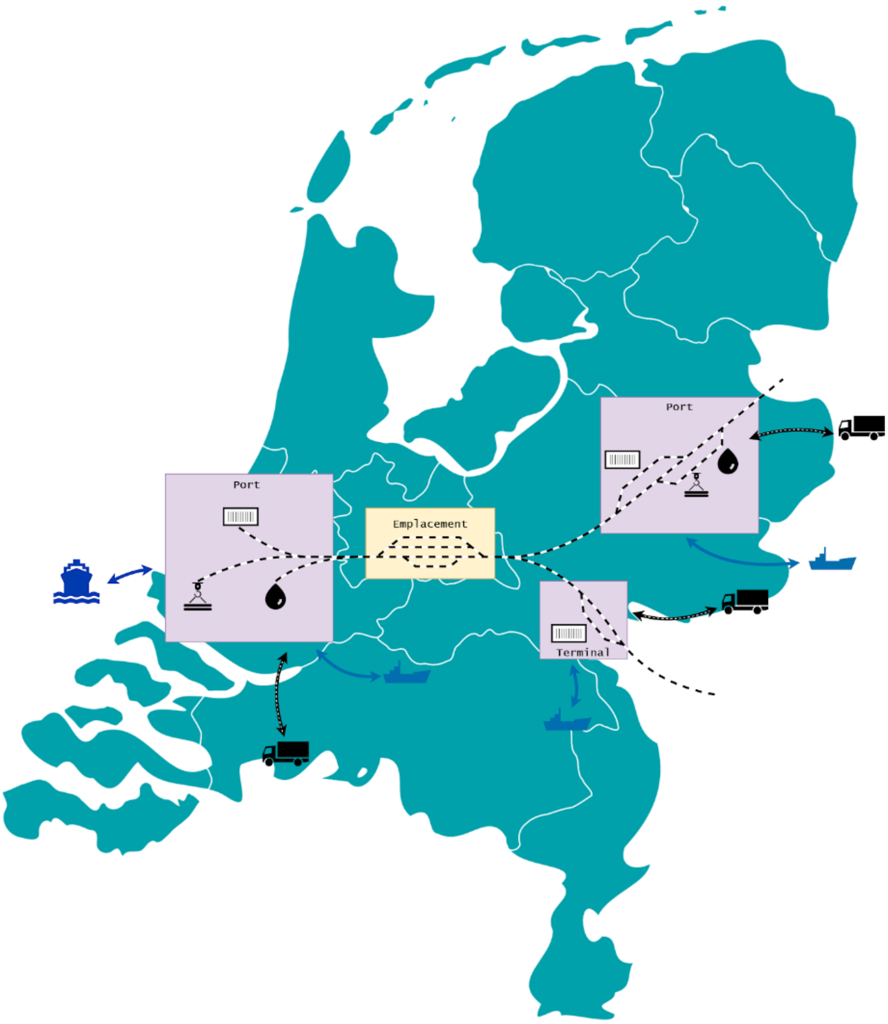


Figure 2.5: Rail freight network

When the freight is loaded, cars are sent to a yard in which they are classified and grouped in blocks: one or more blocks and an engine unit together form a train (Fernández L., 2004). The blocks forming a train, depart and arrive at an emplacement. For all cargo, the destination is known, and through a complex network of parallel railways, the wagons are ordered and attached to the locomotive that travels to all destinations: the other ports and terminals. Once past an inland port or terminal, freight destined to that specific location is transhipped, new freight might be added for upcoming destinations and the train travels further to fulfil all orders. Empty cars are cleaned, inspected and made available again.

Even though few freight routes exist, these routes do not always provide the most efficient paths. This was backed up in the interview with the rail Carrier, in which it was said that for many purposes the mixed net, that is the network used for both cargo and passenger hauling, is of better use. Having two completely different types of transport on the same network could imply some extended complexity for delivering timetables for the Network Operator. For ensuring proper service and avoiding conflicting timetables in the mixed network prioritisation is needed.

2.3 The foundations of Digital Twins

When it comes to accurate data synchronization, digital twins have often come up as solutions in manufacturing industries. This section explains in short that a digital twin is the digital representation of a physical object. Having such a realistic model of an existing object or process comes with major benefits of the transparency it provides, such as remote operation and predictive maintenance. It will be shown that digital twins have a wide-ranging purpose, by giving theoretical classifications of digital twins, using different perspectives. Furthermore, this section will explain the scientifically grounded digital twin architecture by Redelinghuys (2020) that will be used as a baseline for the design proposed in this thesis (Chapter 7). Finally, some examples of digital twins currently in use in the railway sector will be presented.

2.3.1 Definition of a “Digital twin”

The concept of a digital twin is to have a digital representation of a real-life situation, such as machines, products or processes. An example of an early digital twin is found at NASA, in which space capsules were rebuilt digitally to mirror the physical object digitally and perform full simulations of flights in orbit (Surianarayanan, 2020). NASA's system is one of the first examples of a digital twin, the concept of which was first mentioned in 2002 by Grieves (2016). Many definitions of digital twins have come up, some more complex than others, but it comes down to having a virtual representation of a physical asset, or set of assets. The Stargel (2012) definition of a digital twin is:

> *“a multiphysics, multiscale, probabilistic, ultra-fidelity simulation that reflects, in a timely manner, the state of a corresponding twin based on the historical data, real-time sensor data, and physical model”.*

Surianarayanan (2020) makes this quote somewhat more understandable by stating the essence of a digital twin cleanly and concisely:

> *“a centralized entity gathering data from multiple sources to supply right and relevant knowledge of the corresponding physical twin”.*

Digital twins exist in many forms, with many purposes and with different benefits. But the overall aspect of a digital twin is captured in the given minimal requirements by the same authors: to include a model that represents the physical object visually, fed by data of the object it is twinning and mapped with a one-to-one correspondence of the physical and virtual object.

Tao (2019) contribute to these minimum requirements by stating three layers a digital twin should involve: physical-physical collaboration in which multiple assets communicate and work together, virtual-virtual collaboration in which multiple digital models can be connected to a network and virtual-physical collaboration in which the physical object could be optimized through the virtual model. When correctly designed a digital twin could contribute to three powerful tools human processes according to Grieves (2014): conceptualization, collaboration and comparison. Where systems enable fast and extensive processing of data volumes, IT does not possess cognitive tools. This is where digital twins could contribute by enabling the cooperation between man and machine. In this section, some practice examples are summarized laying out their respective benefits.

2.3.2 Digital Twin classification

Classification of digital twins can be done through several perspectives found in the literature. Surianarayanan et al. (2020) make a distinction between digital twins based on twin characteristics. They found four different types of digital twins, based on what they represent:

- At the concrete, detailed level we have digital twins of **components**. This is a twin of a single component, say a screw or any construction material.
- Then there is the **asset** level digital twin, in which an entire asset is modelled (train, motor, car).
- Following is the **system** level, in which an entire system of assets is modelled. This could be an entire factory of modelled machines as assets.
- Finally, on the most abstract level, they found the **process** level. This digital twin provides a business view of business operations across the enterprise to measure progress and performance.

Tao (2019) base their classification on what phase the digital twins are used to gain their benefits.

1. First, there are the **product** digital twins that represent the products themselves. This could be used for designing products: testing product designs and materials even before the products have been manufactured.
2. Then there are the **production** digital twins that are used at the manufacturing stage of products. They help producers to control production, track conditions and create precise plannings
3. Finally, there are **performance** digital twins. These are used to track the performance of objects, provide insights into performance indicators and support predictive maintenance.

Kritzinger (2018) also made their classification and based their classes on the level of integration:

- **Digital model**: A digital twin of a physical product in which data flows are not automated. The model is visually, systematically and mathematically identical to its twinned object, but no data synchronization between the two models exist. These could be used for simulations of to-be scenarios, for example.
- **Digital shadow**: these twins are replicas of the physical model and go one step further than the model in the sense that a one-way communication flow of data is

realized. The digital shadow replicates the physical object visually but also conditionally. These shadows are good for implantations of monitoring purposes.
- **Digital twin**: in the most integrated phase the data flows are two-fold. This is where the digital twins could also control their physical counterparts. Or even, in composed digital twins, systems and products could communicate and control one another. This level of integration enables the synergy of systems, where assets could work together holistically rather than having only individual awareness.

Then there are the maturity levels of digital twins, also provided by (Surianarayanan, 2020). Based on the scope of the digital twins they mapped a level onto the use cases they have found:

- **Partial** is the level at which the digital twin is limitedly fed with data. These digital twins are aimed at answering a few specific questions and measuring a few pre-defined KPIs.
- At the more complex level, there are **clone** digital twins, which gather data of plentiful resources. In this case, it is not necessary to pre-define your research questions, but analysis can be performed on everything that is known about the object.
- Finally, the most complex level of maturity is the **augmented** digital twin. This type of twin exceeds the object-specific sensed data. These digital twins are also fed with data of other objects and resources and are shaped by the use of AI techniques.

2.3.3 Applications and benefits

The digital twin has many different applications in practice, as could also be deducted from the different types of classifications. Surianarayanan (2020), Redelinghuys (2019) and Tao (2019) listed different types of applications that were found in different industries. Examples of applications are listed to provide an example of how organisations could benefit from digital twins.

- **Design**: First the manufacturers are using digital twins for the design of their products. By creating a simulation of a product before actually producing it the manufacturer can test different designs, materials and aesthetics and save on misproductions.
- **Optimization:** By tracking all movements within a system, and seeing how components in a system or nodes in a network interact bottlenecks and congestions could be identified sooner and processes could be made more efficient.
- **Remote monitoring:** Having a virtual twin of a physical object comes with the possibility of monitoring what happens in a remote office instead of on-site. Managing multiple processes at once is enabled by having a single location dedicated to monitoring systems at once.
- **Operation:** The twin could serve as a common operational picture (COP). This is a system that allows on-scene and off-scene personnel to have synchronous information and makes that all operating staff is aware of the same event-driven insights. (Wolbers, 2013).
- **Condition tracking:** Digital twins are also often used for diagnostic purposes, for instance, product or machine health management. By twinning machines or products, organisations can keep track of asset conditions and perform predictive maintenance. This could save organizations from malfunctioning machines or save costs in replacing material too soon.
- **Simulation**: Simulations of entire systems could be implemented using digital twins, with all assets working together. Even the production of products can be simulated using virtual models. This could support planning, it could improve

efficiency in the entire twinned system and it could support operational education of staff.

- **Integration**: With many different assets digitized using sensors new possibilities open up. Where assets used to be standalone, entire systems of assets can be virtualized providing an opportunity for communication flows and harmonizing the system. If assets can communicate and work together, they can warn one another, or automatically respond to unforeseen events.

2.3.4 Architecture

How do digital twins work, and why are they emerging now? The latter is most certainly influenced by the Internet of Things: we live in the era of the Internet of Everything, in which physical objects are connected to the network (F. Tao, 2019). All of these IoT (Internet of Things) devices are broadcasted to the network (device clouds) using APIs, interfaces through which calls can be made. This is exactly how digital twins work, mapping all sensed data of physical objects to a virtual object to enable cyber-physical integration. With mapping the objects to a virtual twin we have three main parts: the physical object in real life, the virtual object digitally and the connection between the two (Grieves, 2014). Zhang (2017) came with two other dimensions that have to be modelled regarding digital twins: data and service. This implies that when one is designing a digital twin, one also must think of the data that needs to be gathered and how the service could contribute to business processes. The technology of digital twins has exceeded its own IoT boundaries and started to merge with other technologies such as AI, which supports smart automated analysis and decision making (Stargel, 2012) (Surianarayanan, 2020).

Redelinghuys (2020) came with a six-layer architecture method for designing digital twins. In this paper a framework was created based on the 5-C implementation architecture of Lee (2015):

- Connection level: the network of sensors and actuators
- Conversion level: convert raw data into understandable data
- Cyber level: the models of the twin and database
- Cognition level: using data for simulation, emulation and diagnostics
- Configuration level: have the twin automatically self-adjust based on data insights

Redelinghuys (2020) translated these five levels into a framework for implementing a digital twin from the physical object to the twin. In addition to the architecture, we map the Model-View-Controller paradigm to the framework. This design type splits an application into objects (object-oriented) with classes that take responsibility for three operations: defining the data object (model), displaying the program in a clear view (View) and the connecting piece that translates data into actions and vice versa (Controller). Using this three-way connection of an application makes for better standardisation in which components do not need to know about the implementation of one another (Krasner, 1988). The model-view-controller architecture which is based on the Redelinghuys six-layer architecture is displayed in Figure 2.6: Implementation architecture of digital twins. The wagon is the object being twinned in this explanation. In the physical layer (layers 1 and 2 of the Redelinghuys architecture) the wagon is fitted with sensors and actuators, such as GPS, thermometers and switches. These devices in layer 1 are fitted with their controllers such as microcontrollers and PLCs (Programmable Logic Controllers). For gathering industrial data, standardisation has been developed: Open Platform Communication (OPC). Using this standardized technique, data of all the sensors are gathered through corresponding device drivers and are stored in defined data formats on a secure server, the OPC server in layer 3.

Here one piece is found missing in the Redelinghuys framework: applications often use more than one data source. Imagine the case in which the weather could be

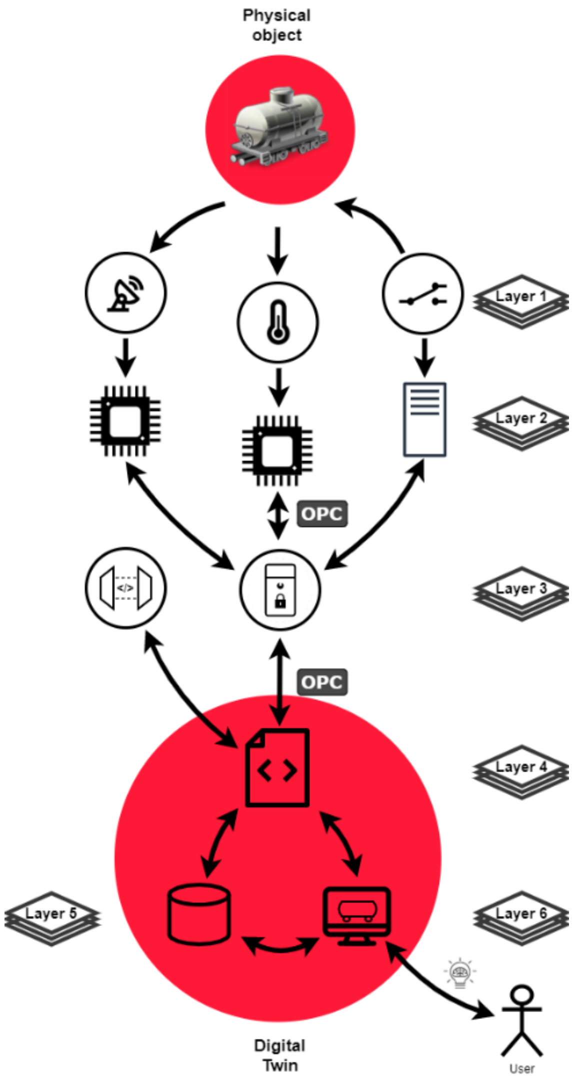


Figure 2.6: Implementation architecture of digital twins

useful for tracking real-time conditions. Then, besides the sensor data and data provided by internal systems, also third-party APIs could be useful. Therefore we added a new element: the end-to-end APIs of servers other than the OPC servers.

In layer 4 is where Redelinghuys assigned the IoT Gateway, a programmed application gathering data from different OPC servers and transforming this data into understandable logic (the Conversion level of the 5C framework). In our model, we replace the IoT gateway with the controller of the MVC application, the part of the application calling data and storing it in the database (Layer 5) through queries. In some instances, this still could be the IoT gateway. The controller scrapes the data of the OPC servers using the OPC library dedicated to the corresponding programming language of the application (PHP, JAVA, JavaScript, C#, etc. all have their libraries). Finally, the data is rendered in the view by the controller using the data of the models. The View is the portal between the user and the entire backbone of the network. Through this display, the user could assign values to actuators based on the knowledge it has gained from the sensors.

2.3.5 Visualization

How the virtual twin is displayed could take many forms, but is dependent on the requirements of the system. In some cases, a 2D form of the object is sufficient, when it comes to tracking and tracing for instance. Or even more basic views could exist if the twin is on a more abstract level, such as business processes, in which plain dashboards will suffice. But there are also instances in which the model of the twinned object is in detail level replicated into a 3D visualisation. Even examples of augmented reality are coming into existence, in which digital twins enter the world of mixed reality are (Schroeder, 2016). This enables new functions of design or training for example.

2.3.6 *Digital twins in the rail industry*

Digital twins are found in many industries, in different forms with different purposes. What about the rail industry? The first thing that comes to mind when thinking of a digital twin in this industry is tracking and tracing trains. “Treinposities”, “Spoorkaart” and “Treinradar” are 2D digital twins displaying all trains in real-time and showing their whereabouts. But directly noticeable is the absence of the freight trains; the betuwe route is even completely left out of the picture in all these examples. These *partial digital shadows* are on a system level in which a *system* of assets is displayed. The phase in which the twins find their benefits are in the *performance* stage.

At the rail level, we see also asset-based digital twins for maintenance of the network. Using sensors and tools such as lidar, GPS and cameras are for generating 3D scans of the rail network, an Italian company is making a fully accurate and detailed digital twin of the rail infrastructure (Nunix, 2020). More of these predictive maintenance focused digital twins are found, such as “Willowrail”, a digital twin of all rail assets. Willowrail is an *augmented digital model* on the *asset* level, in which smart predictive maintenance is enabled. This twin is also in the *performance* phase.

Alstom is also developing a digital for predictive maintenance but is more aimed at maintenance of fleet maintenance decision support (Anylogic, 2020). The twin also supports the design of new railway yards and differentiates itself from that feature from other twins found in this industry. Whereof the fleet maintenance feature the twin could be classified the same as Willowrail, the latter feature of the design is adding an extra phase into the picture: the twin is also used in the *product* phase.

With the focus on virtual testing, we see a solution provided to ProRail by Siemens (Siemens, 2020). Where testing normally is restricted to a specific timeframe in which trains are out of service, Siemens enables four tests to be performed anytime in a virtual environment: track-train integration tests, operational tests, stress tests and testing the onboard units. The last type of test supports the implementation of the ERTMS system by simulating signals passed on by trains. This is an example of the *production* phase, in which operations are simulated and tested. It is a *system*-wide *digital model* of the train network, with rails and trains as assets. Its maturity is likely a *clone* in which the reality is cloned with plentiful data.

On the individual train level, we find the “virtuele trein”, an information system made by CGI commissioned by NS (TIBCO, 2017). A consultant was interviewed about this interpretation of a digital twin in Appendix F. “De virtuele trein”, or the virtual train, was not designed to be a digital twin, but serves as the backbone of many systems including the “treinradar”. The virtual train is a system that gathers information on all different types of trains with different generations of technology to provide a standardized backend for end-user applications. These applications are, among many others, real-time monitoring of trains, dynamic boarding information for passengers and signalling warnings for Train Operators. On an architectural level: trains have many sensors and actuators which are being translated towards internal systems through hubs. Trains have their datastores and a data warehouse is on-shore. The virtual train is not the digital twin itself, it is the first three layers of what could be a digital twin. Other applications use data of the virtual train for implementing its features, fed by other resources. Such as the boarding application providing information about the train itself through the virtual train, but are also fed by track sensors measuring occupancy rates of wagons or check-in systems at platforms. This proves the usable additional resources that the Redelinghuys model was missing. Digital twins build upon this platform have an abundance of information and are therefore at least in the *clone* maturity level.

3 Research methods

In this chapter, some important and relevant research methodologies will be explained. Answering all the research questions and generating a design requires the use of some scientifically grounded methods. The methodologies mapped to the corresponding research phases have been summarized in Table 3.1. The three phases of design research are shown on the left, with their corresponding selected methodologies on the right.

Phase	Selected methodologies
Problem investigation	Design problem template
	Literature review
	Semi-structured interviews
	UML (Sequence diagram)
	TOGAF (ArchiMate)
Treatment design	Requirement specification (User story template and system requirement template)
	UML (Class diagram, Activity diagram)
	TOGAF (ArchiMate)
	Lean
	Development
Treatment validation	Expert opinion
	Single-case mechanism experiment
	Descriptive-, Abductive- and Analogic inferencing

Table 3.1: Methodologies by research phase

3.1 Design research

By default, research that proposes a design is slightly different to traditional statistical researches. Where in many known types of research statistical theories are used to test hypotheses, hypotheses in the so-called design science field are more to a theoretical and abstract degree. This section explains the methodologies of the design science discipline and other applicable frameworks and shapes these themes into a research design that paves the path to knowledge in this thesis.

The basis of the thesis is embodied by the design science methodology of Wieringa (2014). In his book, Wieringa outlies a thorough framework that is designed to increase the scientific nature of modelling so-called artifacts. Artifacts are treatments to design problems: an artifact could be a model, a method, a framework, a process, a piece of software, and so on. Design science is the design and investigation of the artifact in its natural context: the knowledge context and the social context. The artifact is tested in specific cases, and methods are used to generalize the case for use in broader contexts. Testing the design entails the observation of interactions between the artifact and the context. Designed artifacts may show different outcomes in different contexts, therefore it is of significance that the context is clearly stated. In the social context are the involved stakeholders, the beneficiaries of the designed artifact. They provide stakeholder goals and allocate a budget to the design project. On the other side is the foundation of the research: its knowledge context. This is where the theoretical foundation of the project is positioned, providing knowledge about previous research, facts, data, and expert views on certain affairs. The theoretical framework provides the design with ideas and inspiration and gets new information in return for the outcomes of design science.

3.2 Engineering- and empirical cycles

Design science makes use of an engineering cycle in which the treatment, or artifact, is investigated, designed, and validated. The design cycle is shown on the upper right in Figure 3.1. After validation, the new cycle begins in which new problems have arisen or the treatment did not suffice. The new knowledge is used to alter the design and to validate once again. As soon as the validation has been successful the treatment could be implemented and the cycle starts again as soon as it is ready to be upscaled. The cycle accommodates researchers with stages on which they have to focus in what order:

- Problem investigation: What phenomena must be improved? Why?
- Treatment design: Design one or more artifacts that could treat the problem.
- Treatment validation: Would these designs treat the problem?
- Treatment implementation: Treat the problem with one of the designed artifacts.
- Implementation evaluation: How successful has the treatment been? This may be the start of a new iteration through the engineering cycle.

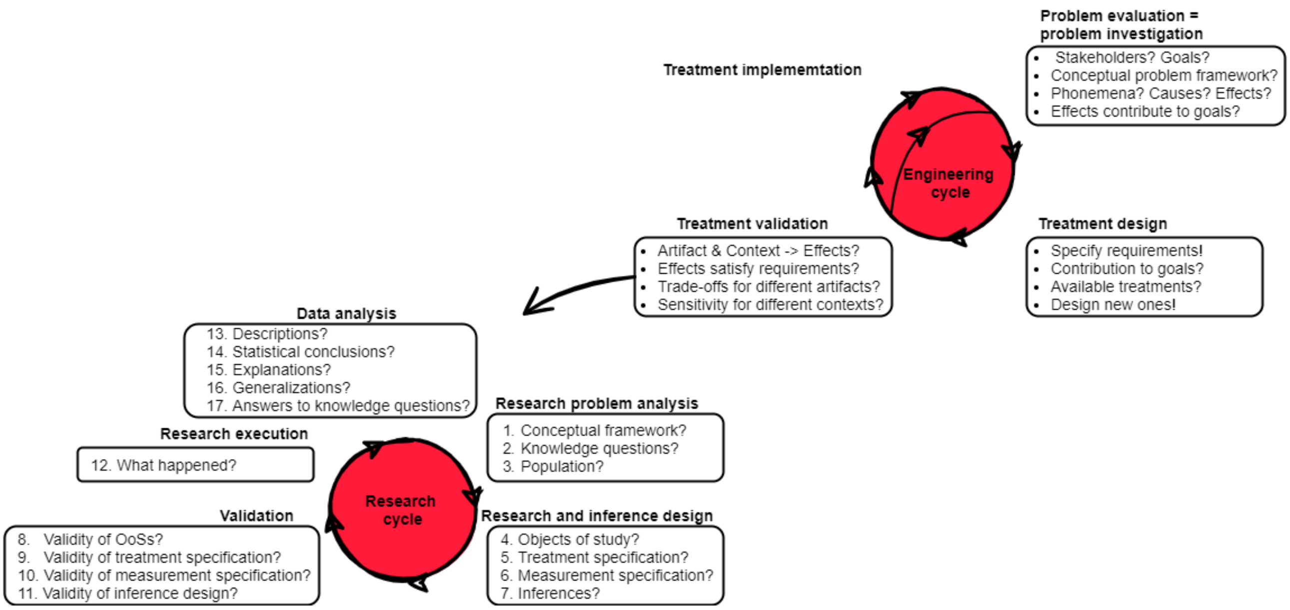


Figure 3.1: The engineering- and empirical cycles (Wieringa, 2014)

Where the engineering cycle is based on the scope of designing the artifact, there is also the empirical cycle for researching the artifact in its context. The empirical cycle is shown on the bottom left in Figure 3.1. The cycle as contributed by Wieringa (2014) shows a rational way for how research in the design science field is to be set up. In this research both are being used: for the design of the artifact we will use the framework as provided by the engineering cycle, and for scientifically testing the artifact we will use the empirical cycle methodology.

3.3 High-level research design

A checklist for all aspects the research design should embrace is given in Figure 3.2, the design of this design research. This checklist has been pruned to suit our thesis and has been filled in with the high-level thesis design. The figure will be elaborated. The overall foundation of the thesis is the design problem. Where the exact design problem will be formulated after a thorough problem investigation in Chapter 6, we can already make an estimation of what the design problem might entail, based on the problem definition as provided in Section 1.2. In this section, it was expressed that many problems in the field were related to transparency and the upward trends of rail capacity. Therefore, at this stage, we could hypothesize the design problem as: "Improve transparency by designing a digital twin in order to increase capacity in rail freight."

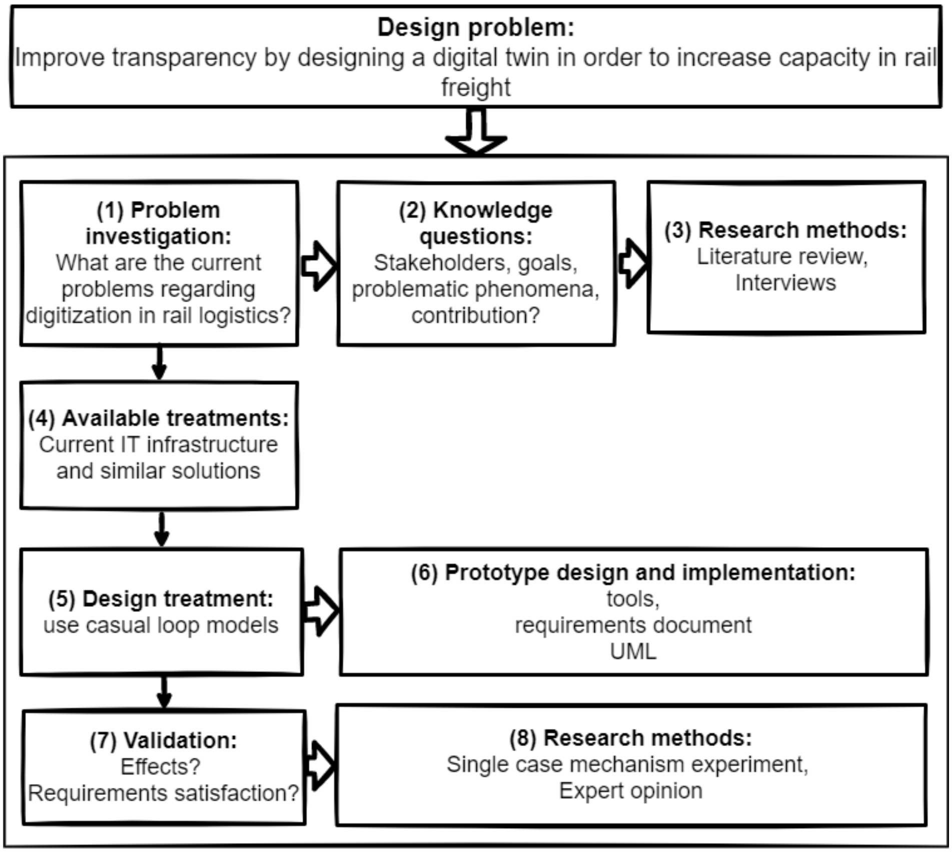


Figure 3.2: High-level research design

The first three steps are regarding the problem investigation and are covered in Chapters 4, 5 and 6. The problem investigation (1) is performed by answering knowledge questions (2) by conducting literature review and interviews as our methodology (3). The outcome of the problem investigation is the design problem as given in Chapter 6. The relevant applications and solutions (4) are partly covered in Sections 2.3.6 and 5.1 and will be summarized in Section 7.3. These available treatments will show the deficiencies in logistics science that could be solved by specifying a new treatment (5). The design of the treatment will be covered in Chapter 7, where the methodologies as described in 3.5 are applied (6). Finally, the design needs to be validated as delineated by the empirical cycle (Figure 3.1) (7). These validations will be performed by applying the selected methodologies (8) as described in Chapter 8 and Section 3.7.1.

3.4 Problem investigation

3.4.1 Design problems

Defining the design problem requires a profound understanding of the field, of mechanisms in systems, of stakeholder goals and missions, of current solutions, etcetera. These knowledge gaps are investigated in the problem investigation phase. The research goal in problem investigation is to identify a design problem which will be formulated in the Wieringa (2014) template:

- Improve <a problem context>
- By designing <an artifact>
- That satisfies <stakeholder goals>
- In order to <help stakeholder achieve some goals>

The design problem will form the framework of the design to be built upon, and for the design to deliver its value to.

3.4.2 Literature reviews

A literature review is aimed at gathering information from different empirical studies that have been conducted using different methodologies and in different contexts (Watson & Webster, 2002). According to Knopf (2006), a literature review consists of two parts: first a summarization of the findings and claims that support the research questions, and secondly an assessment on how correct and complete the information is. The former will form the mainline of the story as all sub-questions will have an element of summarized claims out of literature. The latter will also be a very present element in this study as all conclusions will be backed up by the use of multiple sources, interviews, statistical data, etcetera. Five benefits of conducting a literature review were specified by Knopf (2006):

- It could provide you with a general overview of a topic you are not familiar with (in this thesis the background information in Chapter 2)
- It could help to find existent solutions and therefore not to reinvent the wheel (element used for SQ2 and SQ5 in Section 1.4)
- It could give new ideas for own research
- It could identify where there are flaws in current research (emphasized in SQ3 in Section 1.4)
- It could help to place the research in a larger context

Kitchenham et al. (2004) came up with five steps for performing a systematic literature review in the Computer Science field. This research finds itself partly in this research field, and therefore these five steps were identified as suitable for this paper. The five steps are:

1. Convert the need for information to answerable questions
2. Find the best evidence to answer these questions
3. Assess the validity of the evidence (the use of different sources, interviews and statistical data)
4. Integrate the evidence with software engineering expertise and with stakeholder values
5. Seek ways to improve steps 1-4 (This is where design science takes over)

3.4.3 Interviews

Interviews provide the ability to gather information in a focussed way: they enable data generation by asking specific questions about the matter. Interviews are in several modes (Mann, 2016):

- Structured interviews: interviews in the form of questionnaires in which questions are pre-formulated and fixed
- Semi-structured interviews: questions are pre-formulated but merely act as a script. During the interview, the conversation could deviate strongly
- Unstructured interviews: no questions are pre-formulated, but conversational themes have been defined. These open-ended interactions allow the interviewee to talk at length about a theme.

Structured interviews allow for quantitative information to be gathered for assessing relations between data, such as the demographical impact on cultural beliefs. On the other hand, unstructured interviews are energy-consuming and allow for qualitative data to be gathered. These interviews allow for specific views and thoughts on a matter.

For this research, we have found that semi-structured interviews are most applicable as a methodology. There are specific data we need to know, but this data is qualitative. Insights in the chain are needed as provided by the key players, therefore a script is needed. Unstructured interviews allow for only a few themes to be handled and would therefore also not suffice.

All participants will be asked if they agree with being recorded. If so, they are promised to stay anonymous in every way, and that they could mention whether some information should remain confidential. Recording interviews allow for the researcher to stay focussed on keeping the interview going fluently and controlled, as the need for hurried writing is diminished.

3.5 Treatment design

Treatment design is in itself an intensive task in which several techniques are to be applied. Some of the designs will be defined through requirements: textual statements that describe the functionalities of a system on many fronts. Even though textual representations on themselves could potentially specify entire systems, they do not do such a good job visualizing the system or the relationships between objects in the system. Hence, modelling techniques, such as those provided by the Unified Modelling Language and TOGAF, are needed to illustrate architectural design choices. For the design of the artifact, we have defined a process that starts with high-level design choices that will be the foundations of more detailed design choices (shown in Figure 3.3). In each phase, we will have an agile feedback loop for validating our correctness. First, an artifact will be designed on a higher level: abstract stakeholder goals will be deduced. After the capture of the environment and stakeholder requirements, sub-systems could be defined through the ArchiMate model. This will be high-level: only relationships between stakeholders, subsystems, applications and hardware will be shown first. This should also include relationships between the system and other systems in the environment, as connected systems should be investigated thoroughly to understand their communication interfaces. We do need to understand that implementing such a system is undoable in the timeframe reserved for this thesis. Therefore, implementation will be to the extent of validating the system. The methods as proposed in Figure 3.3 will be assessed in the following paragraphs correspondingly.

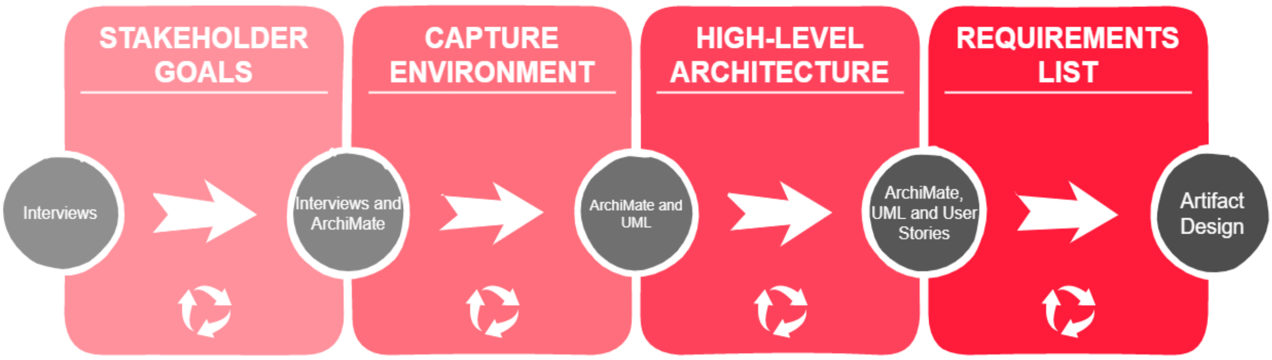


Figure 3.3: Process of designing the artifact

3.5.1 Requirements engineering

A requirement is defined by IEEE (1998) as:

> *"A statement that identifies a product or process operational, functional, or design characteristic or constraint, which is unambiguous, testable or measurable, and necessary for product or process acceptability (by consumers or internal quality assurance guidelines)."*

In short, a requirement is a textual representation of a single feature. Lauesen (2002) defines several levels of requirements: on the highest level, you have the goal-level requirements that specify what a system must achieve, but not how. Then there are domain-level requirements that specify tasks that need to be supported to achieve the goals. This is followed by product-level requirements that specify what processes support the tasks without going into design details. The last is specified in the in-depth design-level requirements.

In this thesis, there will be a focus on goal level requirements, for defining how the artifact could contribute to the stakeholders. Then there will be a focus on product-level requirements for defining the high-level components that the architecture should support. And at the lowest level, the thesis will have a focus on the design-level requirements for specifying the specific details of the designs. For the domain- and design-level requirements the template as provided by Dick and Hull (2017) will be used:

The <system> shall <function> <performance condition>.

Another widely accepted writing template for requirements are the user stories as defined by Cohn (2004):

As a [stakeholder] I want to [function] in order to [benefit].

This kind of template fulfils the use of both stakeholder goal requirements and system requirements at once. Because of its simple structure, the user story template has been adopted by many agile software developers (Lucassen, Dalpiaz, van der Werf, & Brinkkermper, 2016). These user stories will be utilized for defining the system and its proof-of-concept in this research.

The design of the system that will be specified will be based on stakeholder goals, as suggested by the design science cycles. These stakeholder goals are best described through the user stories: user stories explain the goal of the stakeholder in relation to a requirement. On the grounds of these user stories, the system will be designed through the methodologies specified in the following sections. These designs will provide the minimum system requirements that could best be specified in the system requirement template: “The system shall …”.

3.5.2 UML

The Unified Modelling Language is the 1996 update of the Unified Method by Rumbaugh et al. (1999). Rumbaugh et al. (1996) created a framework to act as a format of standardisation in Object-Oriented programming. The framework offers several types of models for different purposes in the creation of software: data schemes, collaboration diagrams of objects, information sequence diagrams. Besides the technical models for designing software, UML also incorporates the business views in some models, such as the use case diagrams. These diagrams support the software developer in visualizing and clarifying the design choices. Where requirements are the textual foundation of the system, modelling techniques are very useful for getting the information through to other developers or stakeholders.

Arguably one of the most common models of UML is the class diagram (Figure 3.4). Class diagrams show objects in a system and display the relations between them. This property of the class diagram makes it perfectly suitable for designing relational databases, but due to its simplicity, it will find its use for many different purposes other than database design.

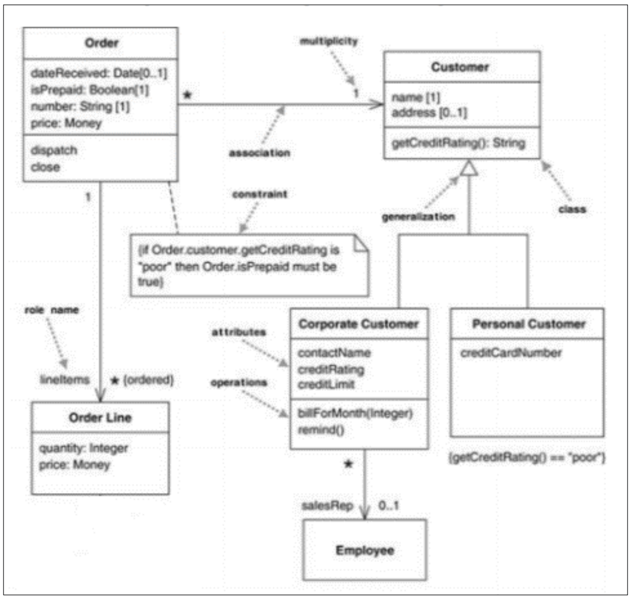


Figure 3.4: Class diagram (Fowler, 2003)

Interaction between components in a system is best visualized using sequence diagrams (Figure 3.5). These kinds of diagrams show the behaviour of different components in a system (users, processes, function, etc.) and will find their use perfectly in describing the communication flows in structures. Sequence diagrams often show events of smaller scenarios, such as the order of a product, but could be useful for displaying flows in larger systems as well. Even though this model is elaborated in this section, sequence diagrams will be used for showing interactions between nodes in the logistics chain in the problem investigation phase.

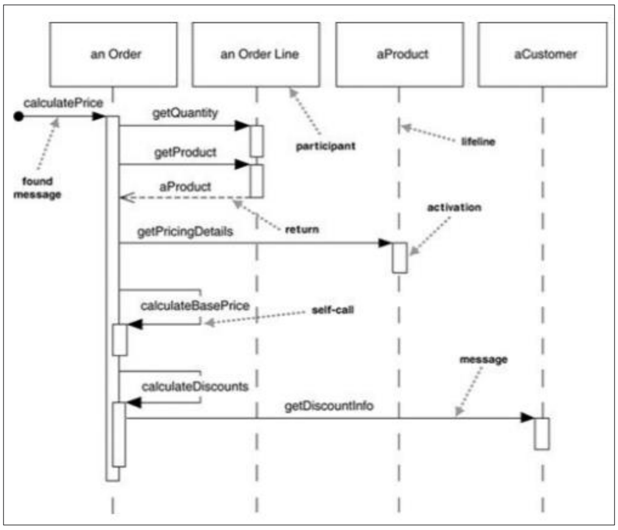


Figure 3.5: Sequence diagram (Fowler, 2003)

Regarding UML, the last diagram being considered in the thesis are the activity diagrams (Figure 3.6). Activity diagrams show the logic behind mechanisms and functions. They are optimized to show the workflow of processes in which a series of tasks and events is shown from A to B. In the flow of the activities in the process, there could also be conditions displayed to show alternative routes to fulfil a task or sequence of tasks.

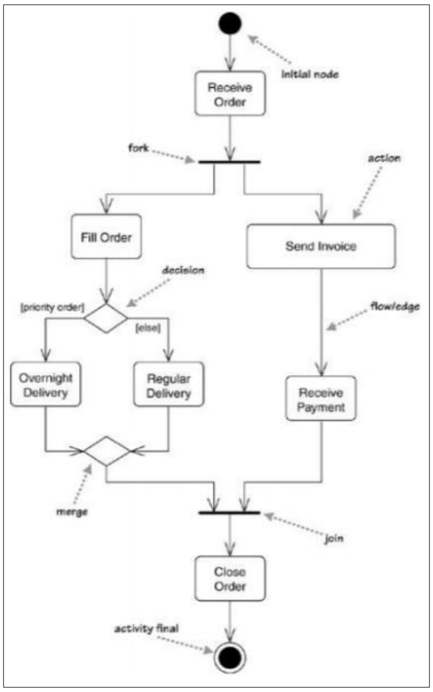


Figure 3.6: Activity diagram (Fowler, 2003)

3.5.3 TOGAF and ArchiMate

TOGAF is an architecture framework for mapping hardware, software, business and actors (Josey A. , 2016). At the core of the TOGAF framework is the architecture development method (ADM) that consists of eight phases as displayed in Figure 3.7. As is shown in the model, the full cycle is built around requirements management: identifying, defining, storing and feeding business requirements. We have the UML modelling for defining most of the behavioural aspects of the system, and even though some architectural models of the UML frameworks are used, most of the architecture will be displayed (in an abstract manner) using methods provided by TOGAF.

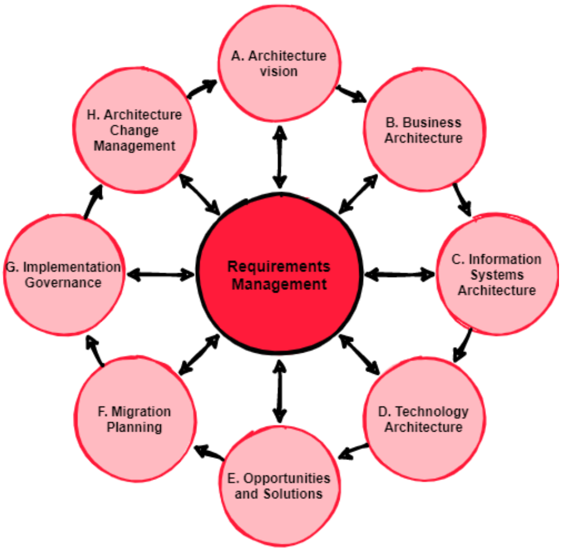


Figure 3.7: ADM cycle of TOGAF (Josey, Lankhorst, Band, Jonkers, & Quartel, 2016)

The ArchiMate framework will be used extensively in this research for capturing and designing IT architectures. The beauty of ArchiMate is in its meta-like character: the developer could specify the needed level of abstraction. ArchiMate could be used to model IT systems of different companies, and how they operate with hardware and users. Or ArchiMate could be used to specify sub-systems of an application and the relation between different users of the particular systems. Or ArchiMate could be used to do both: through different so-called views, the architect could first define the blueprint at the highest level to show different applications in a network to position a new piece of software, and in a more detailed view to show the software's architecture on function level. ArchiMate even shows the possibility to link the application's features to their corresponding requirements, and corporate views and missions (Aldea, et al., 2015).

Figure 3.8 shows the meta-model of ArchiMate: the six levels and their corresponding aspects are shown. We could map this to the TOGAF framework: Phase H and A belong to the Strategy Layer, phases B, C and D correspond with the Business layer, Application layer and the Technology layer, respectively. The physical layer is new but is best mapped with phase D as well. Finally, phases E, F and G belong to the implementation and migration layer. To show Figure 3.8 in relation to the ArchiMate modelling language, Figure 3.9 shows how we should interpret the framework. In the passive structure, we see objects (left in the image), for instance, data objects. In the middle of the figure, we see the behaviour aspect. Many of these behavioural aspects will already be captured through UML, but in a more high-level manner, ArchiMate could be used to show the positioning of these behavioural structures. On the right hand side are the active structure aspects: the subjects that display actual behaviour (such as a user, or application).

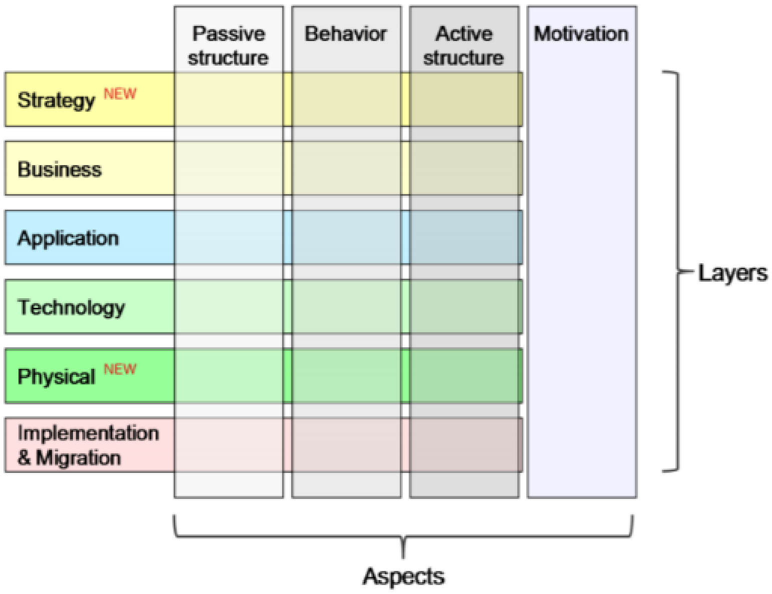


Figure 3.8: Archimate meta-model (Josey, Lankhorst, Band, Jonkers, & Quartel, 2016)

ArchiMate will be used to model the architecture of the digital twin, where the focus will be mainly on the application and the technology layers. However, the modelling language will also play a significant role in blueprinting the IT landscape of the logistics chain, which is positioned in the problem investigation phase. This blueprint will focus less on the technology layer but will show how the business layer is correlated to the application layer.

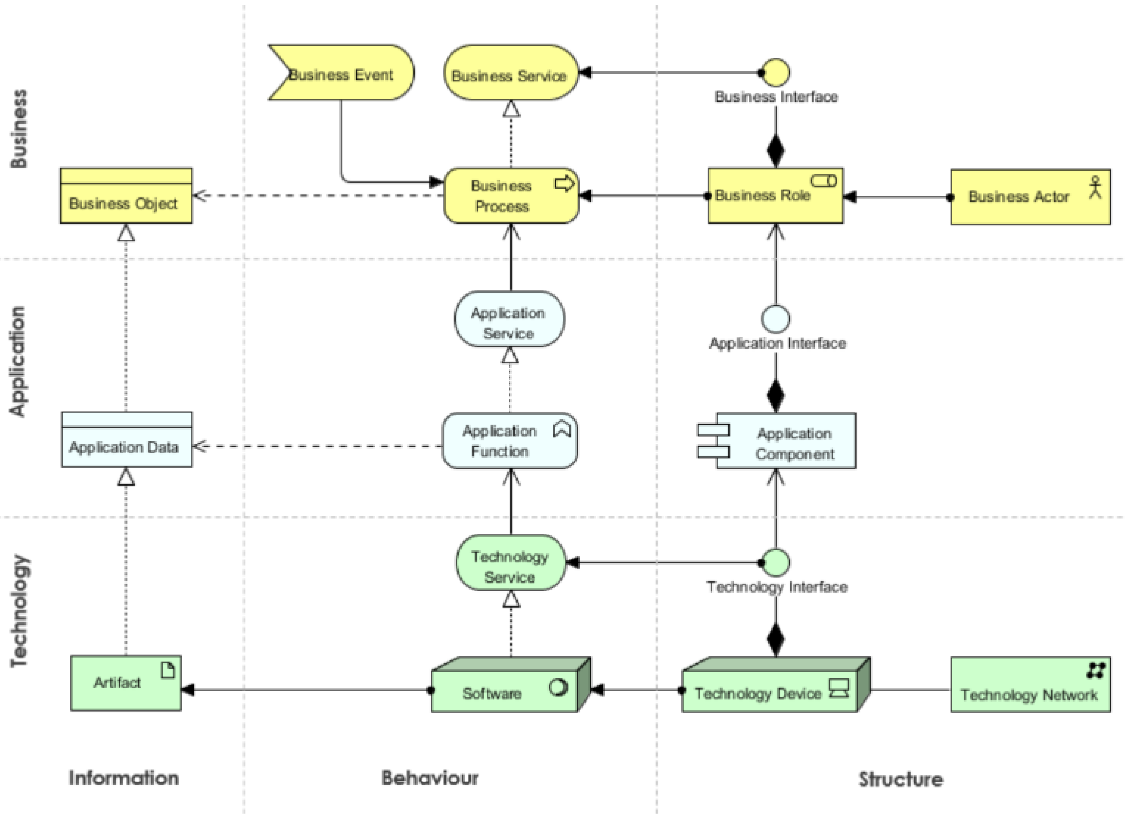


Figure 3.9: The ArchiMate modelling language (https://archimate.visual-paradigm.com/)

3.6 Guiding principles: Lean and MVP

A methodology for designing, implementing and scaling the artifact is of importance and should be considered thoughtfully. A perspective that will be the point of view all made decisions are the principles of Lean innovation and Minimum Viable Product. These two terms will be the leading principles for the design.

The lean innovation philosophy is about targeting the specific problem without overdoing it and without investing in senseless innovations (Sonnenberg & Sehested, 2011). Lean innovation helps organisations by first “doing it right”, then “doing it” and finally “doing it better”. We need to define a (small set of) specific problem(s) for which the digital twin would be most useful and feasible (i.e. “doing it right”). This should be the starting point of the innovation, on which a system could be designed (i.e. “doing it”).

“Waste reducing” is a key definition in lean, because lean is about systematically eliminating non-value-adding features by continuous improvement (Sonnenberg & Sehested, 2011). With the concept of having as little waste as possible, it is straightforward to think of the concept of a Minimum Viable Product. Moogk (2012) researched to show the importance of the MVP concept in lean startups. The MVP should support only the fundamental problem definition and should be made release-ready as soon as possible, to start the improvement cycles early on. In line with these lean philosophies, a low fidelity design problem should be defined which would form the foundation on which an MVP could be constructed (further research).

Lean would also be quite compatible with design science, as design science is about scaling up by design. In design science, several cycles through the engineering- and research cycles are performed. On top of that, design science is about scaling up from simulation contexts towards real-life contexts, and further on to implementation. By maintaining the system leanly after implementation, scaling up towards a valuable system for the entire chain could be realized gradually in a manner that is similar to this research: further design science-based researches could be used to design new usable and feasible components for the Digital Twin.

3.7 Treatment validation

3.7.1 Validation studies

The validation of the treatment shows the effects that the treatment has on its context to answer the question of whether the requirements are satisfied. A validation study requires a validation model which is composed of both an analogic model (i.e. an entity that represents the entity of interest) of the artifact and an analogical model of the context. For the validation of the treatments, different types of studies are defined (Wieringa, 2014), such as:

- **Expert Opinion**: This is the simplest way to validate the artifact. A panel of experts is questioned for their opinions about the artifact. The expert panel has knowledge about the context and the analogical model of the context is provided by their knowledge. The model of the artifact is provided by the researcher and the expert forum apply their judgements whether the treatment may, or may not, succeed.
- **Single-Case Mechanism Experiments**: In this type of research a simulation of the real context is modelled and stimuli are tested by applying a model of the artifact onto the simulated context. Mechanisms are tested one by one, rather than testing the artifact in the larger context.
- **Technical Action Research**: In this type of experiment, the level of simulation diminishes as the researcher applies a model of the treatment in real-life contexts. With respect to scaling up the innovation, depreciating the reproduced characteristics of the context is essential.
- **Statistical Difference-Making Experiments**: In this level of validation, the treatments have been applied to many samples of contextual models. By using statistical analyses, the researchers do not necessarily have to understand the mechanisms around the artifact and its context. However, these studies are hard to control and it may be difficult to deduce external influences from the outcomes.

For this research expert opinions and single-case mechanism experiments will be practised for the validation of the system. Expert opinions are a fast and valid way to prove whether the designed artifact could reach its potential in real-life scenarios. The single-case mechanism experiments could be used to prove the use of different components of the system in hypothetical scenarios. These scenarios could be used as a demonstration that the designated requirements do support the achievement of the stakeholder goals.

3.7.1.1 Single-case mechanisms

First, validation should be performed in ideal conditions with single-case mechanism experiments. Figure 3.10 shows this kind of research setup: the artifact as a full design (on the left) is composed out of functionalities that support stakeholder requirements and goals. These are the single case mechanisms that need to be tested to verify a proper design: both the artifact and its context are split up in which both have an interaction that should be observed.

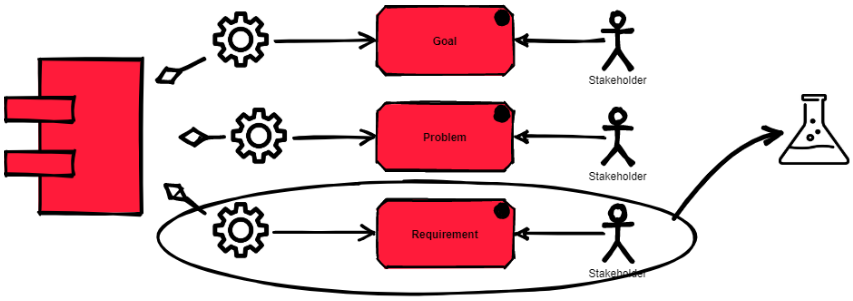


Figure 3.10: Single-case mechanism experiment setup

When performing the single-case mechanism experiments, we first need to define the research problem. This is composed of knowledge questions in the form of the four types of questions as provided by Wieringa (2014) (Section 3.7.1 of this thesis). Then, for testing the effects of the artifact on its context, we need to define the Objects of Study (OoS). The OoS in validation research is the population in which the validation model is positioned. Therefore it is of importance to carefully specify a population that enables generalization out of the inferencing. The validation model is a prototype of the treatment bound to a model of its context, and in this case, it would be one of the modules/components/features of the artifact interacting with the users (the circle in Figure 3.10 represents the OoS). A checklist Wieringa (2014) gave with respect to OoS validity were a matter of inference support, repeatability and ethics. These provide grounds to document about all models: thought must be given to the types of inferences to be made, the repeatability and potential generalization of the study and the care of participating subjects.

3.7.1.2 Expert opinion

In the perspective of scaling up this research starts with idealized conditions, and aims to test the artifact in conditions of practice. This is done by starting with the single-case mechanisms experiment, after which the treatment will be exposed to a fundamental expert opinion study. The scientific nature of this type of validation is in the fact that carefully selected experts possess the knowledge to judge the artifact in their imaginary context of real-world conditions. The experts could tell based on their experiences how an artifact will behave in real-life contexts. This type of validation is efficient and to the point: experts could immediately tell what they might agree with or not. It is critical that the experts are selected thoughtfully: they have to understand the artifact and must be experienced in the specific context. Expert opinions are more flexible in their use, and would not require too much preparation. The researcher should know the specificities of what he wants to know: is it an early verification to tell whether early designs are going in the right direction (the feedback cycles in the stages of Figure 3.3).

3.7.2 Inferencing

We now have the types of validation research we could conduct and the questions we could ask to verify whether the designed treatment complies with its requirements and whether it contributes to the field. The next thing is to observe the studies and disclose the answers to the research questions. Finding knowledge in research is done through inference: making conclusions based on evidence and reasoning (according to the Oxford Dictionary). Wieringa (2014) gives four types of inference, that are also provided in Figure 3.11:

- **Descriptive inference**: summarize data in descriptions
- **Statistical inference**: data is in the form of the population from sample statistics
- **Abductive inference**: to generate plausible explanations of observation
- **Analogic inference**: to generalize your explanation to similar *Objects of Study*

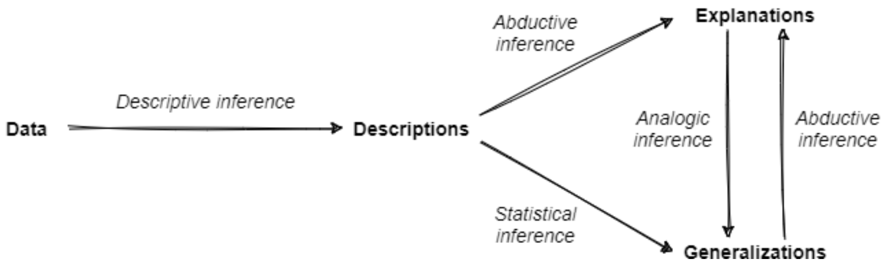


Figure 3.11: Inferencing (Wieringa, 2014)

First raw data is analysed and is given a meaning by descriptive inference. Descriptive data could be used to perform case-based or sample-based research. As mentioned earlier, if statistical validation methods are possible the researcher does not necessarily need to know any working mechanisms and could generalize data through statistical inference. The generalized knowledge could be translated into explanations (of mechanisms) through abductive inference. However, in some cases statistical inference is not possible (or optimal) and therefore abductive inference on descriptive data is needed to explain mechanisms. These mechanisms are generalizable through analogic inference. This process is made clear in Figure 3.11.

The goal of the single case mechanism experiments and the expert opinions in this thesis is to perform *abductive inference*. In abductive inference, we try to explain mechanisms we observe: the OoS (Object of Study, Section 3.7.1.1) dynamics. These are the interactions between OoSs, interactions between OoSs and users and other causal explanations. Causal explanations in this type of experiment are on an architectural level, and therefore are the perfect validation support for testing if our system architectures indeed do perform as they should. Another part of the study is done through *analogic inference*: how the findings of the abductive inferencing could be generalized to the broader scope of the research. For analogic inference, all findings should be analysed to verify the context assumptions and the studied OoSs are to such a similar extent that any generalisation is justifiable.

4 The information sequence

PROBLEM INVESTIGATION

This is the first chapter in the problem investigation phase. Through literature research and interviews, the rail freight chain will be drawn into a sequence diagram to show the flow of information through the sequence of processes. First, actors will be determined and described by their tasks and responsibilities. Then these tasks and responsibilities will be unravelled with the support of the model in Figure 4.1. These processes help to understand what processes are behind the IT infrastructure in Chapter 5.

Cargo flows are barely ad-hoc: they exist in defined and fixed flows serviced by Carriers based on contracts. Rail cargo is just a puzzle in the chain, and goods are transhipped onto trains as door-to-door service is impossible using rails. All over the chain, many parties are included in the process, all having their tasks and responsibilities. Most of these tasks are backed up by supportive IT systems to maximize efficiency. Communication flows in such complex collaboration networks are of importance: it allows for the alignment of all activities in the chain (Bontekoning, 2004). Besides the standard procedures, some incidental events require accurate and timely information. (Den Hengst-Bruggeling, 1999) indicate the complexity of finding liable actors in the case of loss or damage. Also in the case of accidents involving hazardous materials, timely information is needed to be able to react swiftly. What information does each party need to fulfil their purposes? Four interviews are used in this section: two IT consultants working for the Network Operator (Appendices B and C), the Terminal Operator (Appendix D) and the aforementioned rail Carrier (Appendix E). Furthermore, Almotairi (2011) has researched information flows regarding rail logistics. This section briefly describes the important actors and what information needs they have, after which the processes are assessed in detail.

4.1 Actors

4.1.1 Client/Forwarder

The Client is the owner of the cargo and wants it moved from A to B through the most efficient set of routes. Clients are often suppliers such as Shell, Sony or Tata Steel. The Client could also take the form of a Forwarder: this party organizes the shipment from A to B but is not the owner of the cargo, it merely offers a service to the owner of the products. Examples of Forwarders are DB Schenker, DHL or Kuehne + Nagel. Either party asks the Carrier for a specific type of wagon regarding the type of freight. In some cases, the Client or Forwarder itself owns wagons, in which case he only orders the shipment. It was found earlier that cargo flows are settled through contracts, in which the service products for the Clients are defined. The Client requires a volume of freight per year and the rail Carrier serves these needs by defining a product for the Client which is based on its train schedule.

4.1.2 Rail Carrier

The Carrier owns wagons and locomotives and provides Clients with the right types of wagons and transport. Examples are DB Cargo, HSL or Lineas. The Carrier has its timetable which is made in cooperation with the Network Operator. Utilizing this defined train table the rail Carrier goes down a path, picking up and dropping off cargo on the way. Clients always know when the trains will arrive based on the communicated train table. The Carrier needs to know which wagons are ready for dispatch and are most suitable for the Client's cargo and in yards near the Client’s location. This requires an up-to-date overview of the status and location of wagons owned by the rail Carrier. Bontekoning (2004) recognise the complexity of fleet management, due to the separation of wagon and freight (i.e. containers) and due to the many different types of wagons. The rail Carrier that was interviewed achieves good fleet management through a dynamic wagon status.

4.1.3 Train Operator

The Train Operator, often an employee of the Carrier, is responsible for controlling the train. To do so, many checks have to be performed to ensure that the train is safe to go. These checks entail technical inspections, checking the cargo for damage, the order of the train wagons, etc. (Ministerie van Verkeer en Waterstaat, 2004). The Carrier is responsible for handing in a train composition document at the terminal. When a check is performed the Network Operator could give a green signal. At times when cargo is loaded or unloaded at ports, terminals or emplacements, the Train Operator has to redo all checks again. If the train is delayed and fails to meet the given timetable of the Network Operator, then the service has to be requested again. The Train Operator needs to have active communication with the traffic control of the Network Operator to know when he may depart, which routes are made available, alerts about traffic, etc. While shunting the train, the Train Operator has to keep track of all movements in an internal system of the traffic controller. In CBG areas, tracking the trains is done automatically, but in the NCBG areas, this is done manually by the Train Operator. NCBG areas are not controlled using train tabling and require “TijdRuimteSlot” (time and track slots, TRS) requests for entering.

4.1.4 Network Operator

The Network Operator is responsible for maintaining the rail network. In the Netherlands this party is ProRail, in Germany there is DB Netz and in England this is NetworkRail. This party is responsible for rail maintenance, for expanding and improving the network and for traffic control. An active communication exists between the Network Operator and the Train Operator and the rail Carrier for navigating the train. The Network Operator dedicates rail routes to the Carrier and communicates these to them. The Network Operator needs accurate and real-time data of all traffic flows to control traffic as efficiently as possible. This allows maximizing occupancy of the rail network.

4.1.5 Terminal Operator

The Terminal Operator manages all incoming and outgoing cargo flows. Examples of Terminal Operators are The Port of Rotterdam, the Port of Antwerp, or smaller terminals such as Railport Brabant. Many types of cargo from different modalities come together at the terminal, and it has to be managed in what order what cargo gets transhipped to what modality. Planning for transhipment is not necessarily done by the Terminal Operator: the Terminal Operator gets orders for transhipment by the line Carriers. They do need to have the processes aligned with one another, therefore an extent of planning is needed. Especially in the case of container stacking, efficient stacking configurations are needed (Vis & de Koster, 2003). With all the line Carriers of whatever modality they might be, the operator has contractual agreements on what data is exchanged, how it is exchanged and at what times. Especially in the case of the inflexible rail modality, these agreements are very thorough. Terminals have their slots they can dedicate to rail Carriers, in this way they can manage incoming trains and plan transhipment accordingly.

4.1.6 Emergency Service

This is the national service that provides help wherever emergencies arrive, such as fire departments and police departments. Whenever an emergency arises on the tracks, the Network Operator calls the Emergency Service and provides them with track information.

This is needed when polluting substances are on the tracks, for safety management and risk assessments.

4.2 Information sequence

Representing a journey of information within the chain gives an insight into the more technical aspects of the entire process. For finding out how IT supports the chain a thorough understanding is needed of what information is communicated and is needed to prepare upcoming processes in the transfer of goods. The complete diagram as concluded from interviews is to be found in Figure 4.1. This section describes the diagram, which is the first contribution of this thesis. In the figure solid arrows indicate flows, and the dashed arrows are direct responses. In the pools, the different stages of the diagram are presented. Conditional flows are displayed using coloured boxes.

There is no incidental event that triggers the sequence: the flows are defined through contracts, operating patterns and timetables. Therefore it is decided that the sequence starts when the Client requests a product through a market order (A). The rail Carrier responds by designing a suitable product based on the requirements and the train tables. This operating pattern is presented to the customer in the form of a contract that defines flows, volumes, wagon information, needed capacities, prices, etcetera (B). The Client is aware of the timetabling of the rail Carrier and its dedicated wagons are known, information that is forwarded to the terminal (C). The terminal has indicated that they only are executing orders, no transhipment planning is made by them. The rail operators indicate what cargo is transhipped when. As soon as cargo from whatever origin is inbound, they get an EDIFACT-like message (D) containing among others: mass, container numbers, wagon numbers, wagon type, cargo contents, etcetera. This is sent along with a hand-in reference: a document referring to the next shipper of the cargo. These documents are crucial for determining a container stacking configuration.

In train freight it is required to have CIM documents with the cargo: international consignment notes for freight on rail (Comité international des transports, 2017). This consignment note is used for declaring train compositions, which are required to hand to the Network Operator. According to the rail Carrier, it is common that the Client provides the Carrier with this document while booking train capacity (E), however, in some cases, the Carrier generates these documents themselves.

When departure is due, the Carrier sends a request for a train service to the Network Operator (E). Then a train number is assigned to the particular train, which is valid for a single day (F). When the path, departure time and ETA are known, the Carrier requests a timeslot for (un)loading the train at the Terminal Operator through phone or email (G). The Terminal Operator responds with a corresponding reservation (H). The Train Operator gets his "To-Do" list of the train service (saying which locomotive, what paths and stations) by the Carrier (I), which is called a service card. Before departure, the Train Operator has to do a departure check which will be reported to the Carrier (J). As a response, the Carrier forwards the train composition document to the Network Operator (K). The Train Operator can signal the Network Operator when he wants to depart (L), and as soon as everything checks out the Network Operator will give a green light signal (M). If the train of the Carrier is delayed, the service is dropped by the Network Operator and the Carrier has to request a new route using the same train number. Also, the timeslot at the terminal will expire if the train is late and has to be re-requested by the Carrier. This is displayed in the yellow box.

The train has departed and goes onto the "en route" status, in which the train is on the "vrije baan" in the CBG area. Whenever unforeseen events take place causing delays and congestions in the train network, train timetables have to be adjusted (orange box). This means that alternative routes have to be laid in, trains have to wait or even get dropped. This could have an impact on the train of the Carrier and will be presented to the rail Carrier (N). The Carrier forwards the information by updating the service card of the Train Operator (I). Now, within the orange box indicating incidents a red box is displayed which entails the incident being an emergency. In this case, the Network Operator calls the corresponding Emergency Service and provides them with information about wagons on the track (O).

Before arriving at its destination a train often gets shunted at the emplacement. These emplacements are part of NCBG areas which implies that the Network Operator does not have automated tracking of trains and wagons. In NCBG areas it is required to reserve tracks for a certain amount of time through TRS requests (P). The Network Operator gives a free timeslot to the Train Operator (Q). Because the Network Operator does not have a clear vision of what happens on the emplacements, Train Operators are required to manually register their shunting movements (R). In this way, the track compositions are still known to the Network Operator, and in case of an emergency, this information could still be forwarded to the Emergency Services. When leaving the emplacements the procedure stays the same: train checks, check-ins and the green signal if all is right.

Finally, the train arrives at its destination at the terminal. There is a timeslot reserved for the train and the Carrier notifies the terminal that his train is waiting (S). The Train Operator gets a green signal if he is allowed to enter and transhipment could be performed. During transhipment, the containers are checked for damage which would be reported to the Carrier (T).

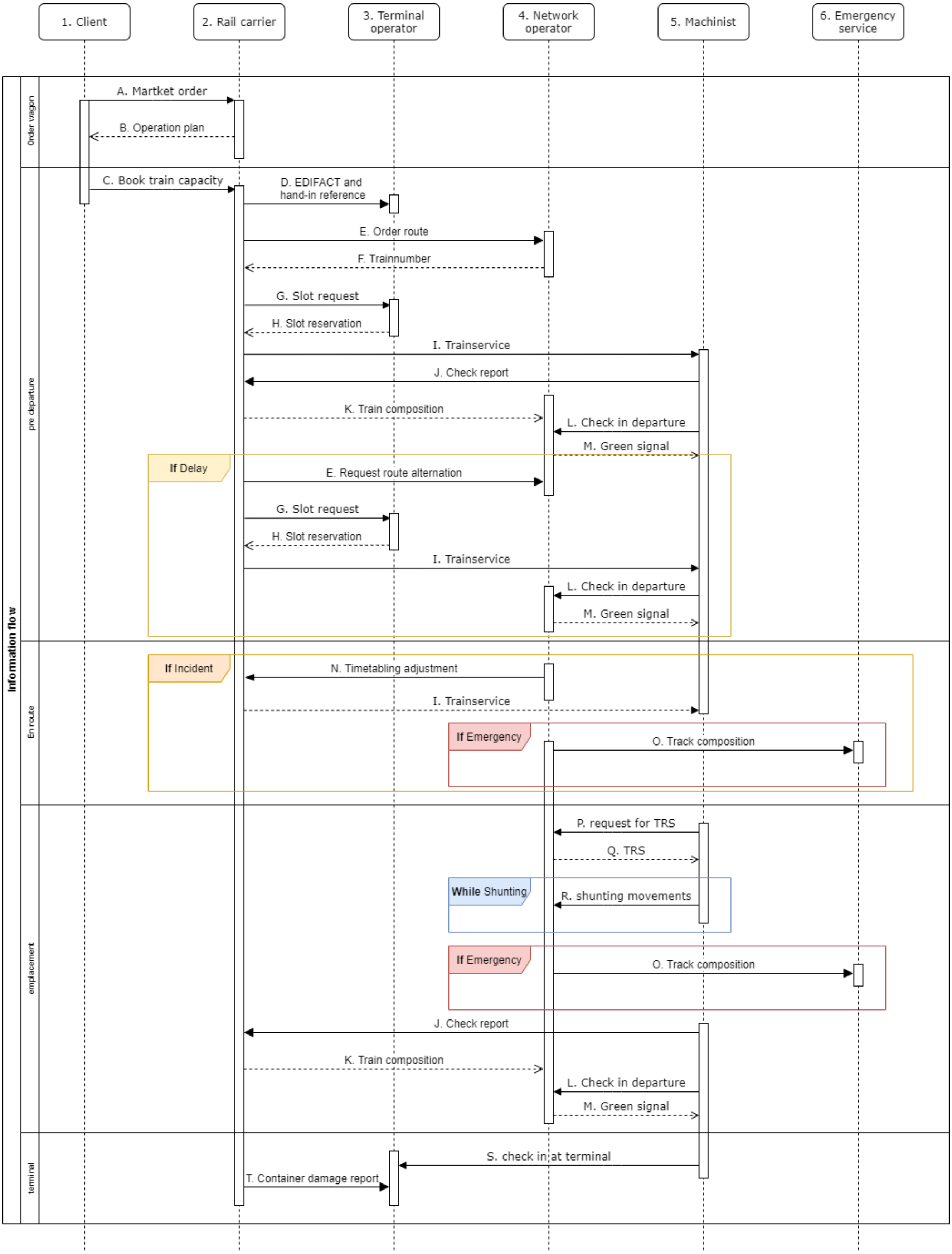


Figure 4.1: Information sequence diagram of transport on rails

5 The IT architecture

PROBLEM INVESTIGATION

This chapter will add the blueprint of the IT infrastructure to the process description of Chapter 4. Many of the processes that are shown in the sequence diagram (Figure 4.1) are relevant to the IT infrastructure and will be mapped to the found IT services in rail freight. In this chapter, first, the literature is scanned and interviews are conducted for finding existing IT systems in rail freight. These are then mapped to the sequences of Figure 4.1, and the users (stakeholders) and Admins (i.e. owners of the software) are determined and listed in Table 3. These findings will be modelled in ArchiMate (Figure 5.1) based on information that interviewees provided.

5.1 Existing IT solutions

In this section, a summary will be made of some IT solutions currently existing in the field, and we try to make a similar classification regarding rail logistics and their direct related operations (e.g. transhipment). By scanning the web for systems and by interviewing the parties a thorough list of over 40 different systems is generated. A total overview of all these systems found is to be found in Appendix A. It became clear that many systems overlapped in their purpose, and it was concluded that some form of generalization was possible. Taking the view of a system's purpose as a classification made it possible to map found software to the actual infrastructure. The following list of different types of software is recognized:

1. **Consignment note systems** for managing the consignment notes of Clients and transporters that are needed for transporting cargo
2. **Emplacement reservation systems** for managing train capacity on emplacements within the NCBG areas. Carriers reserve track sections and the Network Operator manages capacity.
3. **Information management systems** with the sole purpose of sharing information with third parties
4. **Order to cash systems** for managing and administrating entire processes from order to product or service. This type of software could be useful to any party that offers a service.
5. **Path coordination systems** for calculating and ordering routes on the rail network. Carriers order routes and the Network Operator manages the path requests.
6. **Terminal operating systems** for managing all cargo and operation in- and outbound of the terminal
7. **Train detection devices** for detecting whether a train is on a track section
8. **Train dispatching systems** for route insertion for travelling trains
9. **Train information systems** for keeping track of many whereabouts of trains, including all sensed data
10. **Train protection systems** for the protection of trains, such as digital signalling, automated braking and checking whether dispatching is safe
11. **Train timetabling system** for planning and generating train timetables
12. **Wagon information system** for keeping track of many whereabouts of wagons, including all sensed data.

Of the extensive list of systems, only the ones used in the Dutch rail logistics are selected. Table 3 gives an overview of the systems found in the chain, showing which actors are direct users, which flows given in Figure 4.1 are supported and which party is system admin. Dashes in the table indicate that no flow is directly associated with the system, but it does not imply that the systems are not of importance. "Controle app" does not even belong to a type generalization as its use is very specific for this case. It is included, however, as it will play a role in a later stage of this research. The table shows the systems by name on the left, the classified type in the second column, the supported information flow of Figure 4.1 in the third column. The stakeholders that interact with the system are shown in the fourth column and the administrators of the particular system in the last column.

System (producer)	Type	Flow	Actors	Admin
ASTRIS (ProRail)	Train protection system	M	4	4
Axlecounter (Thales)	Train detection device	-	4	4
Controle app (ProRail)	-	-	4	4
Donna (ProRail)	Train timetabling system	-	4	4
DRA (ProRail)	Wagon information system	R	4 5	4
GMS (ProRail)	Information management system	-	2 3 5 6	4
ORFEUS (Raildata)	Consignment note system	C	1 2 3	2
ORMAS (ProRail)	Path coordination system	E	2 4	2
Paloma (Ab Ovo)	Train timetabling system	B, I	2 5	4
PRL (ProRail)	Train dispatching system	M	4	4
RCS (Ab Ovo)	Order to cash system	A, C, J, K	1 2 5	2
RMS/FENIKS (ProRail)	Emplacement reservation system	L	4 5	4
Short circuit (GRS)	Train detection device	-	4	4
Toon (ProRail)	Train information system	-	4	4
TOS (Cofano)	Terminal operating system	D	2 3	3
TROTS (ProRail)	Train information system	-	4	4
UIS (ProRail)	Train information system	-	4	4
VOS (ProRail)	Train timetabling system	F, N	4	4
WI (DBCargo)	Wagon information system	-	4	4
WLIS (ProRail)	Wagon information system	K, O	2 4 6	4

Table 5.1: Systems used in the Dutch rail freight logistics chain

5.2 The IT Infrastructure of rail freight

This section describes the IT architecture of the freight chain as deduced from interviews and literature searches. The model as given in Figure 5.1 visualises the second contribution of this thesis. The colouring of the different elements in the model are according to the ArchiMate modelling language of Section 3.5.3: yellow is the business layer (processes and actors), blue shows the application layer that supports the business layer and green nodes show the technology layer (such as hardware). Some lines are coloured to make the spliced image better readable, no further meaning is to these colours.

A complete model of the IT infrastructure of the Dutch railway logistics created as deduced from the interviews is shown in Appendix G. The model shows to be very complex, is hard to interpret and may contain inaccuracies due to the intricate nature of the overall systems and due to the fact it is based on personal stories. The model should be generalized to a model that is interpretable and applicable to more cases other than the Dutch rail freight. This model does not use system names, but rather uses the generalized type names as stated in the previous section. The model can be found in Figure 5.1. We go through the model based on the information flow of Figure 4.1.

Market orders (A) are contractually digitized and added to an order register or database. The order to cash system is in the case the system serving this feature, however, due to the integrated nature of the system, it was decided to split its features into different applications found in literature and practice that together compose the order to cash system. Creating the operation plan (B) is supported by two train tabling systems: the one of the rail Carrier and the one of the Network Operator. The Network Operator decides on the timetabling and offers routes to the Carrier. The Carrier can then assign operations, wagons, machines and locomotives to its timetable through its timetabling system. Knowing what material is available for dispatching is supported by the train and wagon information systems, that tell about statuses and whereabouts of the assets. Where locomotives often do not have GPS sensors, wagons fitted with these wagon detection devices are more common. Train Operators know about their assigned services through their digital service cards provided by the train tabling system of the Carrier.

Booking routes (C) is supported by booking systems, and the corresponding CIM documents are managed in consignment note information systems. When the Carrier needs to order (E) a route for a dispatching train, the path coordination systems are of service. Path coordination systems serve the train timetabling systems and timetabling adjustment systems for traffic control and capacity management of the Network Operator. The train timetabling system of the Network Operator generates train numbers (F) that are valid for the day. In case of needed timetabling adjustments, the same train numbers will be used. As soon as the train has assigned services by the Network Operator ETAs are known (I) and timeslots can be reserved at the terminal, there are no systems used for this process.

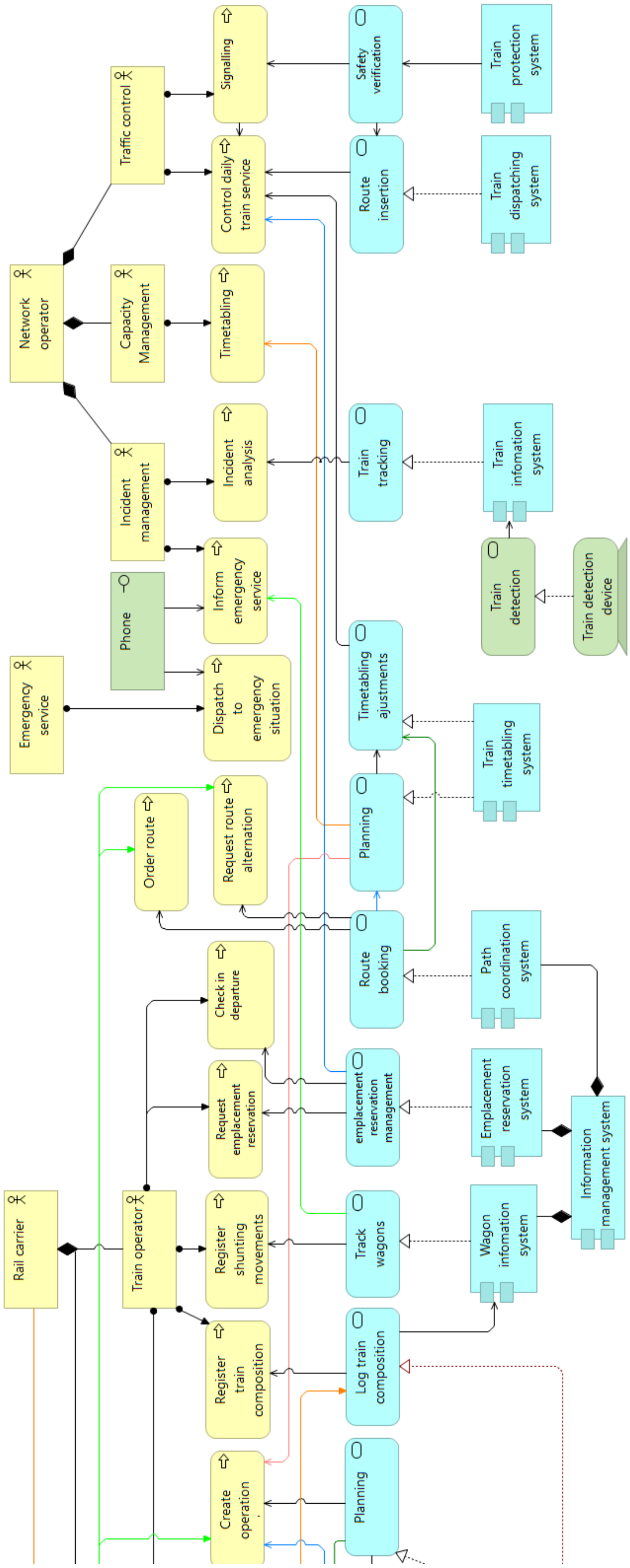
The checks and train compositions uploaded by the Train Operator (J) are stored at the Carrier wagon information systems. These are forwarded to the wagon information system of the Network Operator (K). The Train Operator checks in for departure when leaving NCBG through the emplacement reservation systems (L) and will get a green light signal from the Network Operator. The routes are inserted through train dispatching systems and the train is good to go. These signalling and dispatching systems are served by train protection systems to verify that departure can be performed safely.

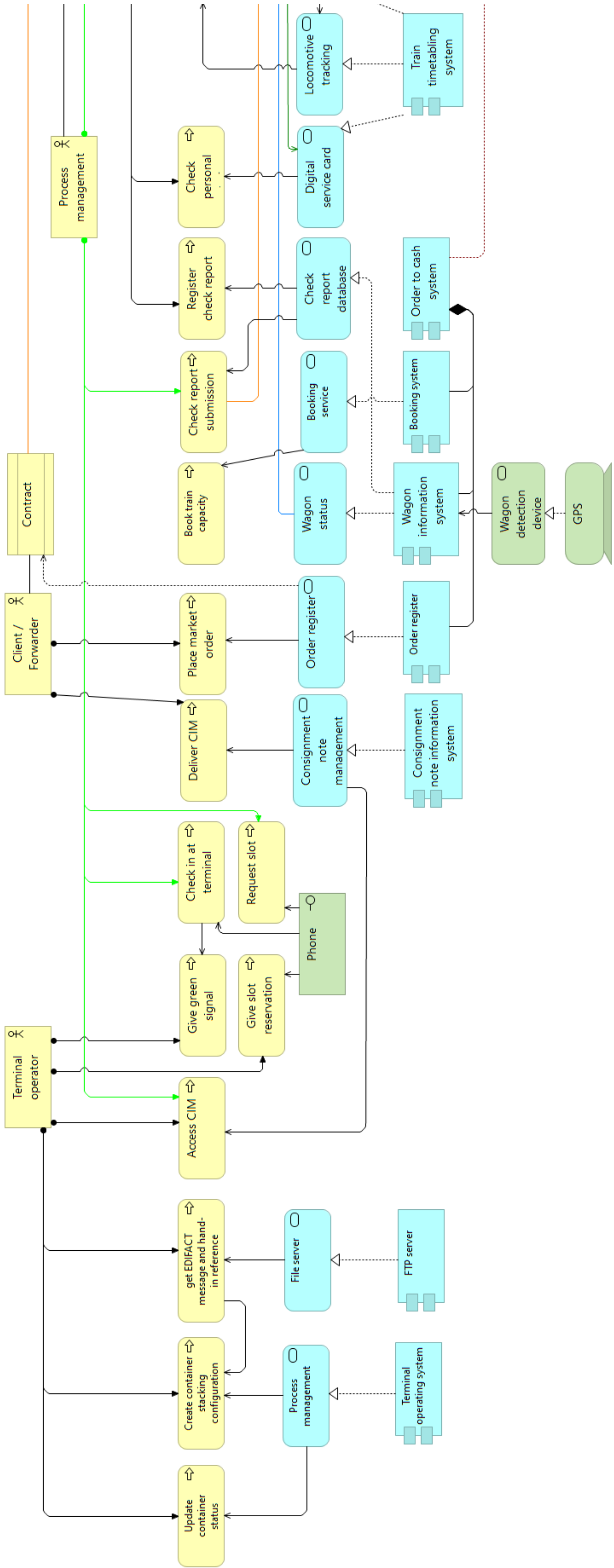
Trains could be delayed and incidents may happen, these give a need for timetabling adjustments (N) that are performed by the train timetabling systems. Routes have to be reinserted and the newly generated train service can be carried out. If the gravity of incidents is large and Emergency Services have to be called, the information about train and track formations is forwarded from the wagon information systems (O). These systems are in CBG updated with train information systems that also tracks the location of trains through train detection devices. In NCBG areas tracking of wagons is done manually in the wagon information systems.

Entering these NCBG areas are often emplacements and for these no centralised management system is applicable. Therefore separate systems are needed for reserving TRS's (P and Q), these are also done in the emplacement reservation systems.

Terminals are found to be quite separated from the architecture as nearly all communication to other parties is done through mail, phone or FTP. Terminals have their terminal operating system which is mostly used for cargo administration: stack configurations, orders for transhipment and all cargo data.

Figure 5.1: The IT architecture of the entire rail freight chain




Process management
Contract
Client / Forwarder
Terminal operator
Check personal
Register check report
Check report submission
Book train capacity
Place market order
Deliver CIM
Check in at terminal
Request slot
Give green signal
Give slot reservation
Access CIM
get EDIFACT message and hand-in reference
Create container stacking configuration
Update container status
Locomotive tracking
Digital service card
Check report database
Booking service
Wagon status
Order register
Consignment note management
Phone
File server
Process management
Train timetabling system
Order to cash system
Booking system
Wagon information system
Order register
Consignment note information system
FTP server
Terminal operating system
Wagon detection device
GPS

6 Formulating the design problem

PROBLEM INVESTIGATION

Based on our findings in the previous two chapters, and the information provided by the interviewees, we can define a set of recurring problems in the field. These problems are often also found in literature and will be backed up by scientific findings. These unfulfilled needs in rail freight will be the basis of narrowing our scope through defining a design problem in Section 6.2.

It is found that IT serves many different purposes in maximizing efficiency in logistics: tracking and tracing, information coordination and automation are the main themes found. The IT infrastructure is developed and complex, but there is plenty of room for improvement. A great number of applications make that much coordination between parties is needed. Port of Moerdijk and ProRail (2020) gathered some information issues in the field that are also found in this section. This section will outline shortcomings of the landscape backed up by what experts in the Netherlands say about the matter, grouped in overarching classes. This is the third contribution of the thesis. Then, the second part of the section will flow into a short summarization of the problem investigation phase and proposes a design problem on which the entire design will be based.

6.1 The shortcomings of the IT infrastructure

6.1.1 Transparency

The most important shortcomings are a matter of transparency. Eliza (2013) find that many of the challenges are concerning visibility: rail agents lack wagon visibility, port operators have restricted visibility of planned rail movements, customers lack the visibility of their intermodal cargo journey and so on. This is understandable, as the rail freight chain includes many parties: there are more than 30 different Carriers, and besides them, there are the Clients, Forwarders, port and Terminal Operators, construction workers, Emergency Services and the Network Operator.

Even though many track, trace and detection systems have been found, IT experts and the Network Operator itself have made very clear that these are not specific enough. Most IT solutions are made for CBG areas, the areas that are neither emplacements nor ports and terminals. However, on the NCBG, where the rail network is most complex, there is no thorough tracking of wagons. Only for wagons containing polluting substances, it is obligatory to register them in the wagon information system. Even though all wagons are displayed in real-time, only the order of the wagons is shown, but not the actual distances of the wagons. This makes looking for the right information on track composition much more difficult in case of emergencies. Carriers do have the information about the freight a locomotive is pulling, but in the journey of transporting cargo, often more than one Carrier is involved. As soon as another Carrier takes over, the full responsibility is taken over. And with Carriers having their systems for this administration, automated tracking of wagons is a very tough task. A test for a wagon identification system is being performed on Moerdijk but shows difficulty.

Portbase has shown to be a big provider of information systems for transparency, but one important link is missing. Where many solutions exist for train compositions, reservations and check-ins, all this information is communicated from Carrier to Terminal Operator by phone or email.

6.1.2 Data accuracy

Besides the lack of information shared between the partners, there is also the matter of trustworthiness and integrity of the data. It is the Forwarder that has to send wagon information to the Terminal Operator, and this data is sometimes found incomplete by a Terminal Operator, especially in cases when cargo has been shipped by many different Carriers. Also, the shunting assistant is completely manually controlled, giving discrepancies in the stored location of a wagon and its actual position. This needs physical verification by using the Controle app. We have seen that this lack of visibility is found a problem in NCBG areas especially, where most data is human-generated and therefore prone to errors. This is backed up by a statement by van Gompel (2016): out of the 193 tested tracks, the composition information (i.e. the wagons and order of wagons on the track) of 75 tracks seemed to be incorrect. To prove this matter even further: the secretary of traffic and environment wrote a letter to the house of representatives stating that 40% of the wagon data in WLIS was falsely registered (Dijksma, 2016). The Terminal Operator has also recognized that sometimes the cargo information is incorrect, which could have large consequences. Especially information concerning next shippers in the hand-in references is sometimes incomplete. Problems regarding position determination of trains are also occurring in CBG areas, GPS data is owned by the locomotive owners (often the Carriers). This makes that locations need to be derived through external sources, which causes locations to be determined on the track-section level.

6.1.3 Utilization

(Gago, 2018) did research in the digitalization of the Polish railway industry. There it was found that some utilization problems exist. They saw inefficient uses of loading spaces, empty runs, non-standardized cargo solutions, traffic disruptions and problems with forecasting. Some problems regarding the utilization of the rail network were also defined by Dutch IT experts (Appendix C). For instance, some emplacements were reserved for four hours when a train was just passing by or using only part of the track. With some kind of central information system, the goal of doubling the cargo shipped on rails is more in reach as this would improve the utilization of the assets.

This utilisation problem has also been shown in Moerdijk and is why tests are performed with automated wagon identification (Appendix C). Because TRS reservations count for a rather large track section for quite a long term (four hours usually), it is often that tracks are blocked for passing trains even though sometimes the tracks aren't even used at that time.

6.1.4 Willingness to cooperate

All the parties mentioned before have to deal with the great complexity of the rail network managed by the Network Operator. This complex chain required the goodwill of all parties involved, to reach proper chain harmonization. And according to experts, this goodwill is not as it should be, as some sort of distrust towards the Network Operator may be existent and competition between Carriers is fierce. The rail Carrier indicated that cooperation with co-Carriers is avoided because they want to keep in control. They also showed no intention of sharing cargo or wagon information unless mandatory, the Network Operator has no right to interfere with any commercial information. In fact, the rail Carrier has shown some frustration towards the IT developments of the Network Operator. When a Train Operator misses its train timetable, the train service is lost and the train number is removed from the systems. This means they have to redo the entire route request process once again. Furthermore, IT solutions are costly and chain partners are not willing to be too invested in the field. The consultants have emphasized the small margins of the transport sector. Gago (2018) also found this unwillingness to cooperate between Polish logistics operators. Cross-chain innovations might have to come from independent fourth parties (4PL). The Terminal Operator showed the level of competition between ports with an outspoken wish to outperform competing ports.

> *"Sharing data? What do we get in return?" – Rail Carrier*

6.1.5 Automation

The level of automation is finding its way with for instance even automation of trains with self-driving train solutions or automatic drayage Carriers. However, where technology advances in these places, in other places such as large emplacements the changes aren't even electronically controlled. In these places, the Train Operators have to manually switch the changes. The self-driving train solutions are ambitious and perhaps a certain future, but impossible to fully achieve as long as environmental awareness is not fully digitally achieved. Before full automation should be even thought of data generation, sensing and automated controlling should be realized. No human-generated data, no manual track changes and no data in silos should exist in this vision.

6.1.6 Liability

Liability is a problem in every cooperation with many players. In logistics it is no different, cargo travels a long way and even though on most nodes in the chain it is supervised thoroughly, many things can happen. First of all, many different Carriers transfer the same cargo, completely transferring liability to the next player in the chain. However, if a Carrier has filled in all documents as handed over by the previous player but the information is not right, he will point to the other parties. It is hard to keep track of where things may go wrong.

Also, theft is a serious problem in NCBG areas. If all parties know what wagon contains what cargo, theft can be easily organized. And who is responsible for cargo is missing at the end of the chain? Who knows when cargo has gone missing. Therefore, transparency should know its limits as there are side dangers.

And finally, where there is AI, there always is a point of discussion about who is liable in cases of incidents. This automation-related subject is in computer ethics often a point of discussion (Čerka, 2015). Say wagon identification is 95 per cent reliable, which sounds very decent. But now imagine that daily over 500 wagons go through emplacement Kijfhoek (ProRail, 2018), which means 25 wagons are missed each day. Whenever emergencies arise and there is no information about a flammable wagon, things could go catastrophically wrong. And who is responsible? This is where the Controle app comes back into the story, and shows to be hard to phase out.

6.2 Design problem

The final section of the problem investigation phase will be the bridge towards the design phase: the findings of the problem investigation phase will be summarized and a conclusive design problem will be formulated. This section will narrow the scope (as given in Section 1.3) within which the artifact will be designed (Chapter 8). The entire design of the artifact as described in the following chapters will be based on the design problem as formulated here.

The problems that stood out the most are regarding the NCBG areas, where transparency and control are lacking. Even though effort has been put into increasing transparency and data accuracy in these areas, the representation of the actual track situations is still sub-optimal: even though track compositions are known, distances between wagons are still unknown. And data is manually generated, therefore prone to errors. With the polluting substances in mind and the fact that many emplacements are in dense areas (cities), transparency should be improved for the sake of risk management.

Also in these NCBG areas, a matter of capacity management has been pointed out as an issue. Lacking visibility and no central service for these areas imply that larger track sections get reserved and blocked for long timeframes. This results in other trains just passing through getting their paths blocked, sometimes unnecessarily. Having a better understanding of what happens in these areas could result in more specific TSR management, in its turn resulting in better capacity allocation for handling more trains in the network, supporting the goal of increasing the use of the rail modality and thus enhancing the possibility of reducing congestions on the road. However, according to the consultants, the scope of the problem could be expanded slightly towards CBG as they emphasize the importance of capacity management. A partial design of a digital shadow would be most feasible to come up with an MVP design (minimal viable product). A digital model won't be useful as no simulation purposes would make the MVP and for reaching purposeful features real-time tracking should be performed. When designing the twin data should only be gathered if it contributes to the problem statement. A 2D map of the rail network will be sufficient for generating enough knowledge for supporting decision making in terms of risk or capacity. Digital Twins could be a solution to the information challenges related to transparency with an emphasis on, but not limited to, NCBG areas to improve capacity management and emergency risk assessment. The design problem to support this issue is:

Improve transparency with an emphasis on NCBG areas by designing an MVP 2D digital twin that satisfies business requirements of chain parties and innovation growth prospects while maintaining data security in order to improve capacity management in freight train service and risk assessments in case of emergencies.

Starting a design should be aimed to be low-cost, therefore a design should have the least possible investments. An MVP should make use of as much data as is already available, such as using existing software. This is backed up by the Dutch ministry who does not want to build its own IT systems for digital transformation in logistics, but would rather see the use of existing IT solutions (FEDeRATED, 2021). In the FEDeRATED article, it is mentioned upon which foundations digital transformation should be built, such as users should be in charge of their data (data sovereignty), trust must be built through security and privacy, standardization measures are needed for interoperability, the platform should be neutral (i.e. independent of any party) and quality of data has to be accomplished.

Identified applications in this field would be train information systems, wagon information systems, emplacement reservation systems, train detection devices and train timetabling systems. With all data currently available in these applications information is known about what cargo is in what wagons, what wagons are on what tracks, which trains carry what wagons, which trains are currently travelling, which wagons are about to depart, what wagons will be entering emplacements and so on.

When thinking about the future more desires have been outspoken. Carriers would like to have more accurate information when carrying handed over cargo, wagons may be followed by GPS data, shunting movements may be assisted automatically by wagon detection devices (but not completely taken over by automation due to liability issues), future flows could be simulated systems as requested by the port area manager by using historical train data or train timetables served by the train timetabling, signalling could be done through transforming the shadow to a twin using the signalling systems, data could be added of follow-up shippers of cargo by using timetabling of other modalities, and so on. But the key to lean and agile innovation is starting low-key.

7 Digital Twin design

TREATMENT DESIGN

We have now covered the problem investigation thoroughly, which is the first phase of the engineering cycle. The outcome of the problem evaluation was the formulated design problem, which will form the basic foundation on which the requirements will be built. In the Wieringa (2014) engineering cycle as depicted in Figure 3.1, the treatment design will follow the problem investigation. This phase of the cycle focuses on the design of the new artifact in the problem context that contributes to the stakeholder goals in the social context and the existing knowledge in the knowledge context. The treatment design should clearly state how the requirements contribute to the stakeholder goals and should clearly document existing solutions. This chapter will follow the structure accordingly (corresponding with Figure 3.3): first, the stakeholders and their goals will be summarized, then we summarize relevant systems and available treatments, after which we will model some high-level design to show an abstract design in relation with its context, and finally all the findings will be summarized into one structured requirements-list. This line of work is a build-up from the low level (the why) towards the detailed design specification (the how).

7.1 Stakeholder groups

In order to define the stakeholder goals, first, the stakeholders have to be determined. In the timeline of this research, several parties have been identified and approached, of which the eight most relevant have been incorporated into the IT architecture as depicted in Figure 5.1. These stakeholders are also shown in Table 7.1. The most present stakeholder is the Network Operator and many internal parties within this organisation have been found active in different processes within the freight chain. Therefore this stakeholder was split into three separate stakeholders/users, as shown in Figure 5.1: Capacity Management (responsible for long term timetabling), Incident Management (responsible for handling unforeseen incidents and emergencies) and Traffic Control (responsible for managing traffic and ad-hoc timetabling). Furthermore, besides the stakeholders belonging to the Network Operator, we have the Carrier (executing the cargo of the freight on the rail network): a stakeholder which has been split into Process Management (responsible for managing train services) and the Train Operators (responsible for executing the cargo transfer). Finally, we have the Clients/Forwarders (the owners of the shipped cargo), the Emergency Services (responsible for handling emergencies on site), and the Terminal Operators (responsible for transshipment of cargo).

7.2 Stakeholder goals

All of these stakeholders find their value in the transparency provided by a digital twin. These values could be phrased as goals according to the user story template as provided in Section 3.5.1, and are shown in Table 7.1 (overleaf). Section 3.5.1 explains the reasoning for using this template.

It was decided to discard some of these goals, for different reasons. First of all goals number 5 and 6. Carriers already perform fleet management and often do so by installing their own GPS sensors (goal 5). They have shown no intention to share this data with the Network Operator, and therefore the position determination should come independent of the Carriers. They could benefit regarding goal number 6, and this might even be an incentive to negotiate them into sharing GPS data. However, technical checks should not be replaced with AI solutions without careful thought! We have

Goal	**1. As Capacity Management (Network Operator) I want to see the shunting behaviour of my Clients (i.e. the Carriers) in order to increase the number of fulfilled capacity requests in my timetable.**
Explanation	Capacity Management has the goal to increase capacity, or in other words: increase the number of fulfilled capacity requests of the Carriers. By analysing shunting behaviour, capacity management could find inefficient shunting movements and unused TRS allocations for instance. The shunting movements could also be used as part of a topic of conversation to resolve conflicting desires of Carriers.
Goal	**2. As Incident Management (Network Operator) I want to see accurate wagon information in order to increase the effectiveness of risk analyses.**
Explanation	Incident Management has the task to respond to incidents on the rails, making risk assessments and analysing these incidents afterwards. By having a clear vision of what happens and also what happened on the network, incident management could inform Emergency Services more accurately, determine how an incident happened and how it could be averted and could decide whether certain wagons could form a risk to densely populated areas (Moonen (2018)).
Goal	**3. As Traffic Control (Network Operator) I want to see actual train positions in order to allow more ad-hoc capacity requests.**
Explanation	Traffic Control has the task to maintain a constant traffic flow on the rails. On NCBG areas visibility is lacking (Section 6.1), which is the reason that the Network Operator uses TRS allocations (Section 4.2). On these TRS areas, the occupied tracks are unknown, and in the case of single track entrances, some TRSs block entrances to other TRSs. When Traffic Control would have visibility of cleared entrances, more ad-hoc requests could be honoured.
Goal	**4. As the Emergency Services I want more accurate wagon information in order to respond more accordingly to specific situations.**
Explanation	Currently, too many wagons are registered wrongly into the Wagon Information System (Section 6.1.2). Therefore, in case of emergencies of a serious nature, the Emergency Services can not completely rely on the given data. By providing a digital view of the as-is situations, the Emergency Services could quickly determine any hazards regarding wagons containing polluting or flammable materials.
Goal	**5. As Process Administration (rail Carrier) I want to know the whereabouts of my wagons and locomotives in order to improve fleet management.**
Explanation	Part of the tasks of Process Administration (Interview in Appendix E) is keeping track of assets, and dispatching these assets in an efficient manner (such as utilizing wagons closest to a call). If the Carrier would be informed about the position and the health of his assets through the system, his fleet management would certainly benefit.
Goal	**6. As the Train Operator (rail Carrier) I want to have real-time asset health visualisation in order to decrease the time to complete technical checks.**
Explanation	Every time Train Drivers depart, they have to perform an extensive list of checks (Section 4.1.2), including wagon health and composition checks. These checks have to be performed quickly, and therefore are prone to errors. By having a digital view of the train composition and asset health, the Train Driver could both reduce checking time and increase the accuracy of the other technical checks.
Goal	**7. As the Client/Forwarder I want to view the current position of my freight in order to increase process alignment.**
Explanation	Clients/Forwarders have decreased visibility of their shipments due to increased intermodality (Section 6.1). If they would have real-time insights into the current location of the freight, they would have the possibility to align their processes better to the arrival of cargo.
Goal	**8. As the Terminal Operator I want to see accurate train compositions in order to prepare transhipment more efficiently.**
Explanation	Terminal Operators always prepare the required machinery for loading or unloading incoming cargo (Interview Appendix D and E). They know the train compositions and place their machinery accordingly even before the train has entered the terminal. Through this system, they could have a clear view of the train composition and location, which would improve the efficiency of preparing transhipment.

Table 7.1: Stakeholder goals

pointed out the complexity of AI-based decision making, and due to these liability concerns in case of AI being wrong, these checks should always be performed by a responsible stakeholder. Hence, goal number 6 was also rejected. Goal number 7 would require the Digital Twin to display sensitive commercial information, something that should be handled carefully. This is also assessed on page 51, where liability concerns have shown to be of relevance in this project. Besides, tracking and tracing of cargo would require different research to be performed, and therefore would not fit the scope of this thesis. And finally, we will discard goal number 8. The benefit as stated in this user story shows potential to the Terminal Operator, but the Terminal Operator has shown no interest in such a system at all and indicates that they rely completely on the direct communication with the Train Operator (Interview in Appendix D). On top of that, also this user story will fall out of the scope of this project.

This mentioned scope, as framed in the design problem in the previous chapter, has been focused on capacity management and risk assessments in rail freight. Goals 1, 2, 3 and 4 perfectly suit these given research borders and are the foundation on which the design will be built. From these assessments, it is concluded that the Network Operator benefits the most from an operational digital twin. Moreover, it was earlier found that the Network Operator is the central entity in the IT landscape (Chapter 5 and Section 7.1), and has the most resources to realise such a system.

7.3 Relevant systems and available treatments

An extensive assessment has already been performed of the current applications in the context of Dutch rail logistics in Chapter 5 and succinct in Section 2.3.6. In this section, we briefly discuss the information that could potentially be used to feed the digital twin.

7.3.1 Moerdijk

Starting with a concept that has been mentioned in Section 6.1 without getting its deserved attention: the Moerdijk setup. At this track site, testing is done with independent wagon recognition, which entails wagon recognition without being dependent on the wagon owners. The issue is known that it is impossible to install sensors on the assets of the wagon owners (often the Carriers), especially because of the international nature of the logistics chain (Moonen H. , Gevaarlijke stoffen juist op het spoor, 2018). Fitting GPS on locomotives would be more feasible, but a train composition does not maintain the same locomotive: especially on NCBG areas the wagons are pushed around by different locomotives. Therefore wagon recognition through RFID, Bluetooth or GPS tracking is not feasible and other solutions have to be found. One of the architects of the proof of concept explained the IT infrastructure behind the tests.

The solution that is being tested at Moerdijk concerns wagon recognition and detection through a complex network of sensors. We first have axle counters to detect whether a wagon is on a certain track section. Then there are sensors to detect the statuses of track switches, to determine the route the wagon is taking. And finally, there are the smart camera's to read the UIC numbers from the wagons. Through this system, it is possible to detect where what specific wagon is positioned. The track occupations are visually displayed using a schematic model of the track network of the emplacement, which is not geographically correct.

7.3.2 Botlek

Besides the Moerdijk setup, this one did not get the required attention either. The Botlek setup is quite similar to the Moerdijk setup, both setups are proof of concepts regarding automated tracking of wagons and trains. The Moerdijk setup is based on the identification of wagons in order to improve data accuracy regarding track compositions. The Botlek setup has a more organisational perspective of data analysis. In this setup data is mostly aimed at improving the network and its capacity: better division of TRS areas, improve bottlenecks on the rails, finding too slow sections, and so on. For this setup, they mostly use axle counters and track switches to determine the direction the trains go or have gone.

7.3.3 TROTS

TROTS is mentioned a few times earlier and it seems to be a key player in the IT infrastructure. Through this system the position of trains is determined, at least that is the essence. What the system does is detect whether a section is occupied. The difference with the Moerdijk setup is that TROTS works binary: either a track section is occupied or not, it does not count how many wagons are positioned somewhere. The focus is on trains rather than on wagons. The determination of what trains are positioned on the occupied track section is done by reading out the train timetables through which it could be ascertained which train is most likely to be at that location.

7.3.4 WLIS

Also, WLIS has come back in the report a few times. In this section, it is made clear why it is of importance for the digital twin. As explained before, WLIS is the system in which track and train compositions are registered. It is known from each train which wagons are being transported, and from the wagons the UIC numbers, responsible Carriers and what contents they entail (at least for the polluting substances). WLIS would be very significant to the twin when combined with the tracking systems of Moerdijk or TROTS. Knowing the UIC of a wagon as an identifier would imply that from a tracked wagon all necessary information could be deduced. Or using train numbers, train compositions could be fetched from WLIS using TROTS.

7.3.5 Better tracking and identifying

Unfortunately, the systems are lacking in their trustworthiness: complaints about the Moerdijk sensor accuracy were stated, and WLIS track compositions did not seem to be accurate either as found earlier. There is a need for improved systems for wagon tracking. In general, WLIS wagon information and track compositions are unreliable. Therefore the most significant successes could be made with improving tracking systems. Perhaps the use of Lidar technology could be used to scan rails on wagon presence, or proximity meters to detect more accurate occupancies. Even drones or satellites might be applicable to perform image recognition for track occupancies.

7.3.6 Asset health detection

Besides the previous applications of detection systems, there is also the one for asset health detection. There are many detection methods for predictive maintenance for the rail infrastructure, but also regarding wagons detection devices are used to measure several aspects of a passing wagon. There are systems to detect wheel profiles, to see whether some wheels are not perfectly round anymore causing extra risks of derailments. Or detection mechanisms to detect axle-loads, whether not too much weight is carried by specific axles. Then there are systems to check the health of brakes, also quite an important aspect when it comes to train safety. And finally, there are systems to detect hotboxing, a term that is used to indicate certain aspects are overheating due to load or resistance. This could have severe consequences for wagons containing dangerous goods, and therefore any overheating of an axle, brake or wheel should be detected early on.

7.4 Design choices

For the high-level architecture of the design, we should first classify the type of twin that would suit the stakeholder goals. For the architectural decisions, we refer to the levels of a digital twin's architecture as shown in Figure 2.6 (page 24). In the following section we first determine the specific classifications (Section 7.4.1) and benefits of the proposed system (Section 7.4.2), after which each of the designs will be provided (Sections 7.4.3, 7.4.4, 7.4.5 and 7.4.6).

7.4.1 Classification

This classification is done using the terms as given in Section 2.3.2. We see a need for a twin that displays several assets: trains, wagons and rails are the minimum needs for the goals. This makes that the designed twin will be on a *system* level, in which all assets are displayed and identified as part of the entire system that composes rail freight. Furthermore, we have the twin on the *production* level, even though this classification is somewhat less relevant. This classification is mostly concerning digital twins with manufacturing purposes but could be justified as that the process of timetabling and asset management has overlap with production processes. A relevant classification is the level of integration: the twin needs to be a *digital shadow***.** This means that there needs to be a one-way connection between the physical object and the digital model. Sensors need to deliver real-time information, but the need for actuators is absent. There is in no situation a need for controlling wagons or trains. Maybe in the far future track changes could be controlled through actuators, but that would be for purposes outside of the scope of our system. Finally, there is the level of maturity that would at least be needed to be able to fulfil its purposes. In line with our MVP reference frame, the lowest level of maturity, a *partial* digital twin, would suffice. If only the data is presented that is necessary, the twin would be of great value. However, it is not the data representation and visualisation that would be a great challenge, but making sure that wagon positioning and identification is reliable.

7.4.2 Benefits

The benefits this twin potentially provides are threefold, given the categorised benefits as given in Section 2.3.2. First and foremost we have *operation* as the main incentive: a twin should make for all parties involved to have synchronous information. In case of an event or incident, the twin in rail logistics provides traffic control, the Carriers, incident management and Emergency Services with the same information to make sure all parties are informed synchronously about the conditions on the rails. We should bear data sovereignty in mind here, not all parties should be able to see sensitive information. Then there is *condition tracking*. We have listed some of the condition tracking devices for issues such as unround wheels, hotboxing issues and detection of overloaded axles. This is useful information for risk assessments regarding hazardous materials for instance. Finally, we have remote monitoring as a great benefit: a digital twin would offer the possibility to perform functional operations from a remote office.

7.4.3 High-level Architecture

On page 23 the layered architecture of Redelinghuys was mapped to the MVC vision in Object-Oriented Programming resulting in the abstract architecture of Figure 2.6. This layered perspective will be the basis of how the digital twin in rail freight is envisioned. A digital twin should be a digital representation of its physical twin (Section 2.3), and how it is visualised is variable to each purpose of use. Digital twins could have extra potential through modularity, expanding technology allows for gradually improving and extending towards new uses, better data acquisition and sensor fusion (i.e. combining data from sensors to create new understandings that are not possible to get from individual sensors). Both the Redelinghuys and the MVC views allow for expansion: the MVC makes that different views could be linked to the model and controller, and the Redelinghuys architecture makes that new sensors and other data sources could be linked to the data server.

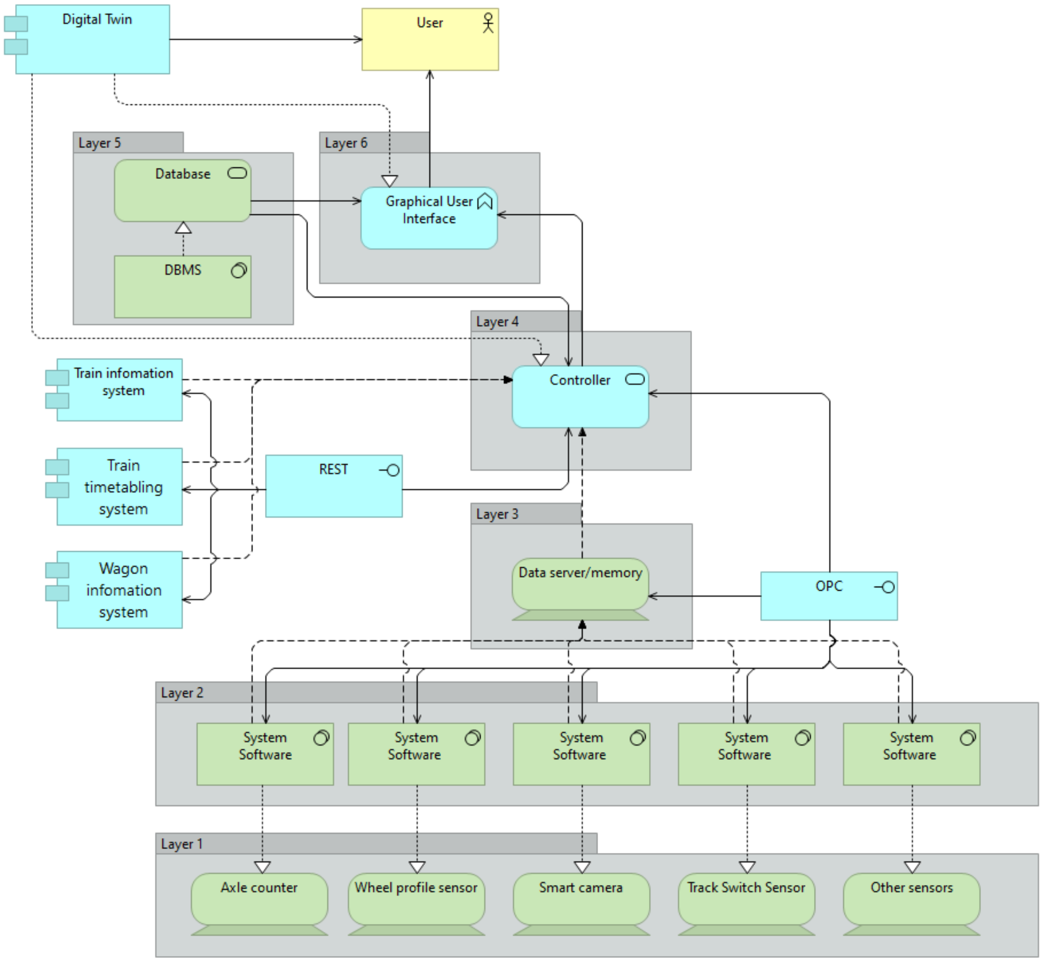


Figure 7.1: High-level digital twin architecture

Implementing the abstract architecture in our context brings us to Figure 2.6. On the top is the user that interacts with the Graphical User Interface: knowledge is derived from the data visualisation. As shown in the figure, the application itself in the MVC structure will be composed of layers 4, 5, and 6. Of the three, layer 6 is the GUI that shows a direct interaction with the user. The controller, in which programmable logic is stored and run, is layer 4. Together with layer 6, layer 4 is directly hosted in the application. Layer 5, the database, is often a separate server (and sometimes the controller and view are also separate servers) that is accessed by the controller and sometimes also by the GUI (however, this is generally indirect access). Data is gathered from the sensors through a network of interfaces, controllers and servers. The controller of the digital twin is connected through an OPC protocol to the data server or memory which is positioned in layer 3. Layer 3 is the gateway to all sensory devices that keep track of the context of the physical object. We have discussed thoroughly the environment of the digital twin itself, with which we point to the surrounding applications such as WLIS and TROTS. These systems are relevant to the dataflows of the digital twin, and should therefore also be incorporated into the architecture. For exchanging information among applications the RESTful API interfaces are common. Through this interface, data is scraped from the applications, which are an addition to the Redelinghuys model. Finally, we have layers 1 and 2, which are respectively the actual physical device and the controllers with the logic to handle incoming and outgoing data.

Automated wagon identification has been tested in multiple proof-of-concepts, examples of which are the Moerdijk and Botlek test setups. The problem of incorrect data in the Wagon Information Systems is due to human errors. Whether AI should replace human responsibility is a sensitive matter and will not be propagated in this thesis, but the supportive potential will be acknowledged. Therefore these setups have been an inspiration to the hardware requirements of this architecture. Sensors that are used in these setups for wagon tracking and identification are Axle counters (to count the number of passed axles), track switch sensors (to detect the direction of track switches) and smart camera's (to read UIC wagon numbers). By incorporating the WLIS and TROTS data, corresponding data could be gathered about track compositions. Cameras could also detect wagon labels to add a digital check to the registered wagon information. Furthermore, there could be future experiments with camera-equipped drones, proximity sensors, LIDAR and other exciting innovations. Keeping these developments in mind we emphasize modularity once again.

7.4.4 Object relationships

After explaining our choices on the layered architecture, we can dive deeper into the MVC components, starting with the model. Some recurring objects have been recognized and could be pinpointed into a relational data scheme. In the MVC principle, the model (layer 5) is composed of objects, that often have some relation to one another. As mentioned: trains, wagons and rails are composing the digital twin in this scope. There is an abundance of digital rail maps and the rails are not the assets we aim to twin in this thesis, and therefore such information should not be incorporated into the objects composing the database. Rail maps could be reused through different libraries and sources. Hence, the focus is on wagons and trains, of which the schema is shown in Figure 7.2. Don't focus on the event table just yet, this will be discussed in the upcoming paragraph.

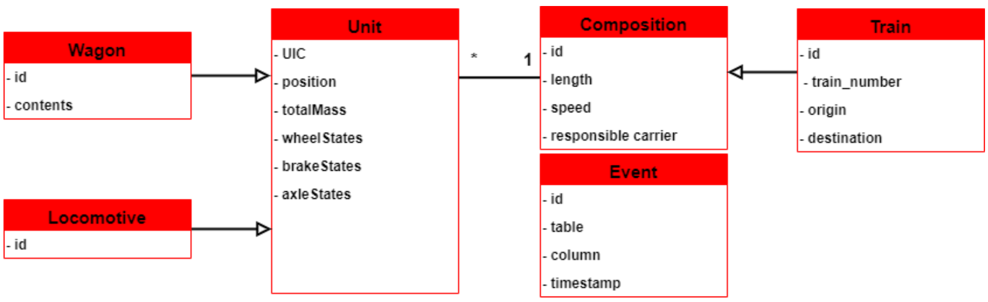


Figure 7.2: Relational database scheme

We have seen that not all track compositions are *trains*: *wagons* are mounted on certain tracks to get loaded, unloaded, shunted or dispatched. There should be a relationship that tells this story, as pointed out with the outlined arrow: a *train* is a *composition*, but a *composition* is not necessarily a *train*. This same "generalization relationship", as they are called, is also seen regarding *wagons* and *locomotives*. A *composition* is built up from *wagons* and *locomotives*, sometimes only wagons and no locomotive and sometimes even multiple locomotives to carry a set of wagons. Therefore we generalize wagons and locomotives into a *unit*: a *composition* is made out of several *units* in the form of *wagons* and *locomotives*. The *unit* field consists of most of the fields, as both generalized types encompass these same characteristics.

The minimum required fields are deduced from the stakeholder goals and some interviews with stakeholders. With respect to capacity management, they are most concerned with data on a train level. They want to know where a train is (position), what train (train number) is driving, how fast a train is driving (speed), where a train comes from (origin), where a train is destined to go (destination), what Carrier is responsible for that train (Carrier), and how long a train is (length).

In the case of incidents and safety, the information is more focussed on the individual wagons. The most important field of wagons would be the UIC: the UIC is the common identifier through which all other sorts of data could get scraped through API calls. Knowing the UIC provides the system with a lot more information than just the identification, but other information would be known as well. The train position as mentioned could be inferred from the position of the individual wagons. Therefore the "position" field is incorporated in the "unit" table. Even though it is important to know the contents of a wagon for safety purposes, actual contents should not be displayed, as this would be sensitive commercial information. Therefore it is decided to display only label information in the system. Finally, there are the three state fields: "AxleStates", "WheelStates" and "BrakeStates". The asset health detection devices provide information about the wagons' wheels, axles and brakes (while remaining the possibility to add more devices). These states are displayed in the "unit" table.

7.4.5 Event storing

This section is focused on more detailed design choices on how the controller of the MVC components should process the data. One table that was not discussed in the previous paragraph is the "event" table. This part of the design is inspired by both the Redelinhuys architecture and the architecture of the Moerdijk setup. Sensors have different ways of registering data: a temperature sensor might have a continuous flow of data while a track switch sensor only registers the change of a state. Therefore, the connection between the controller and the sensors is not bilateral: it is impossible to have a continuous connection that requests data of all layer 2-controllers at a fixed frequency. Therefore the design needs to be rather an SMS-like architecture: a device sends information to a server (layer 3 of Figure 7.1), which buffers it and sends it to the receiver at the time the receiver requests the updates. By serving as a memory, layer 3 enables the possibility for asynchronous information sharing between the sensors and the data model. This is an example of "event sourcing", a method that has been used in the Moerdijk proof of concept. Event-sourcing is a mechanism that stores states as a sequence of events[2]. Layer 3 is the memory in which all of the sensor data is stored. The controller calls for any updates as soon as a connection is made, and the controller transforms the new data into events. These events are mutations of known data, for instance: at 12:04 a smart camera registered a wagon at a certain location, which would be a mutation of the "position" field in the "wagon" table. Figure 7.3 shows our interpretation of such a mechanism which we will explain in the flowing sections.

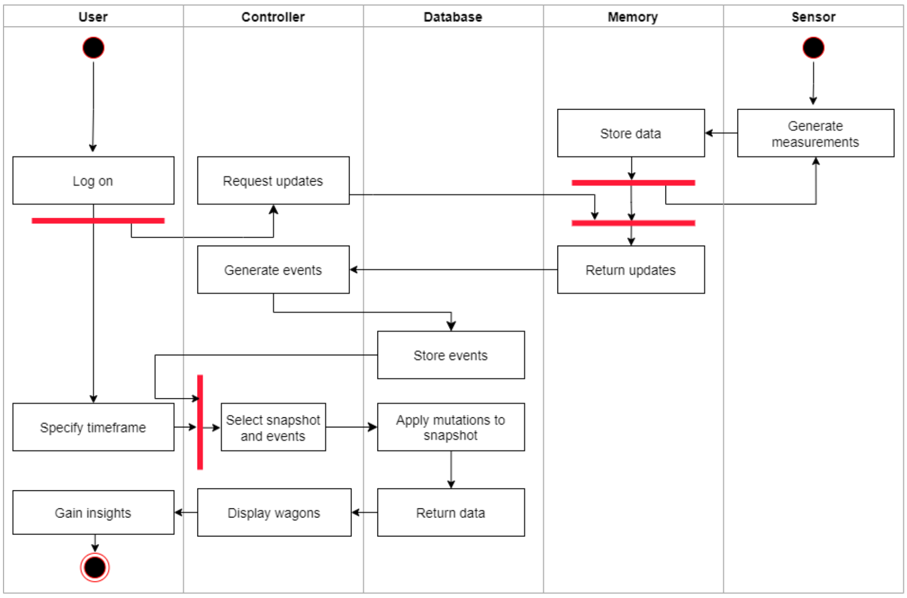


Figure 7.3: Activity diagram of event storing architecture

[2] https://martinfowler.com/eaaDev/EventSourcing.html

At the Moerdijk setup every week a snapshot is made of the current state of the database. In this way, the system could start with a clean slate each week, and when selecting a timeframe the system does not need to process an impossible amount of mutations before having the preferred state. Saving the current state could be done in a plain SQL dump (a script resulting in the build-up of a database with the current tables, columns and data), or in any other compatible way depending on the chosen database technique.

We'll address Figure 7.3 step by step. On the right-hand side, we have the sensor that constantly generates and measurements and sends these to the memory. The red bar indicates a parallel task: the memory both awaits requests and gathers new sensor data. Once a request is inbound, the memory answers the request according, all the while still expecting new incoming sensor data. The memory acts as a buffer in this way, making sure the connection does not need to be bilateral.

On the left side is the user who logs on and gets the system online. As soon as a user logs on and selects a timeframe (i.e. a preferred state of the data), the controller starts preparing the data. It asks for any updates in the memory and if so, it generates events out of these data entries. Events would be the outcome of the controller calculating the required database changes from the new data entries and transforming these into a row consisting of the table and corresponding column to be transformed (as displayed in Figure 7.2: the "event" table). These events will be stored in the database.

Every given period, the database makes a snapshot, to which the events (or mutations) are applied. When the controller has processed all updates, it selects the most recent snapshot and applies all events to the snapshot. The database mutates the tables and gets the database to its desired state. This makes that communication is asynchronous: sensor data could be requested when needed, and no direct connection between the two is required. Furthermore, the snapshot limits the processing time of the event mutations and reduces the impact of missing event data. Finally, the data is returned to the controllers and is made visual by the graphical user interface for the users to gain their insights.

7.4.6 Visualisation

This section assesses the stakeholder goals (Table 7.2, overleaf) to propose a design (Figure 7.4, page 63), as explained in the following paragraphs. Finally, we present the design for the view aspect of the MVC components. Digital Twins are digital representations of physical objects, and accordingly, the visualisation should mirror the physical object. Data only gets its value when presented properly to the users, raw data contains knowledge but is no knowledge on itself. Therefore effort should be put into the visualisation and the representation of the data. Many stakeholders have expressed the need for transparency: knowing the whereabouts of trains and wagons is key in visualisation. Deducing from the stakeholder goals we will need an accurate visualisation of wagon and train positions, fed with information on both train-level and wagon-level.

Therefore: we incorporate both the wagon views and the train views in our design. A 2D visualisation has been hinted upon a few times in this report, and due to its simplistic nature it seamlessly align with the needs of the stakeholders: no intricate 3D correct models are needed to achieve the stakeholder goals.

We'll address each of the incorporated stakeholder goals individually as shown in Table 7.2. From the stakeholder goal assessments of Table 7.2, the mock-up as shown in Figure 7.4 (page 63) arose. The GUI shows a geographical correct view of the railway tracks in the rail network. Note: in the actual system a map of surrounding streets and buildings would be useful, but this has been left out for visibility purposes of the mock-up. There is an option to have either a historical view or the current real-time situation on the tracks. This makes that the need of several users are met through the

Goal	1. As Capacity Management (Network Operator) I want to see the shunting behaviour of my Clients (i.e. the Carriers) in order to increase the number of fulfilled capacity requests in my timetable.
Assessment	This goal is about the analysis of historic train movements. This mostly concerns inefficient shunting movements, inefficient TRS allocation or having a common operational picture for the Carriers and the Network Operator to deliberate upon. These historic movements call for a specific time interval to be selected, which should be one of the requirements. Then, from the selected interval, the ability to play the events in the timeline should be incorporated.
Goal	2. As Incident Management (Network Operator) I want to see accurate wagon information in order to increase the effectiveness of risk analyses.
Assessment	When it comes to incidents on the tracks with the possibility of involved hazardous goods, the essence of the data is based on real-time usage. The focus of these parties is to operate based on risk: instead of either denying or approving incoming goods on emplacements or in densely populated areas, these parties could act more based on solid risk assessments. Therefore, here is once again the need for a wagon-level view in order to assess each wagon individually on their health statuses (whether wheels have been worn, or axles have a risk of hotboxing). If for instance hotboxing occurs in a wagon containing flammable contents, an interference would make sense. The timeframe of the data should both be historic and real-time: real-time for risk projections and historic for the analysis of past situations. Additional important information could be the current location of the fire brigade. There is a usable API served by the government to provide this type of information. This last assessment adds some emphasis to the need for the possibility to connect different API sources.
Goal	3. As Traffic Control (Network Operator) I want to see actual train positions in order to allow more ad-hoc capacity requests.
Assessment	Traffic control would mostly be concerned with the as-is situations. For ensuring a safe and continuous flow of traffic, traffic control is mostly dependant on accurate real-time data. Precise location information of trains is mainline in this story. Therefore, the requirements for real-time data streaming and train detection are of importance here. You could expect from train views, that the user is more zoomed-out than with the wagon views. The system could auto-switch to different views based on the zoom levels.
Goal	4. As the Emergency Services I want more accurate wagon information in order to respond more accordingly to specific situations.
Assessment	It has been stated multiple times that data accuracy is seriously lacking and that the need for an automated system is rising. Wrongful information regarding hazardous materials could be catastrophic. However, we have also found that AI-based data generation should not be leading: liability concerns will arise. Therefore we will not incorporate the use to replace the information registration by the Carriers. But the system should be capable of monitoring the trains and wagons automatically. The importance of the accuracy of data is in the case of incidents: the fire department needs to know about surrounding cargo. Imagine a fire near flammables, or leakages of polluting fluids. Therefore, a clear real-time vision is needed to provide fast information to create contextual awareness.

Table 7.2: Stakeholder goal GUI assessment

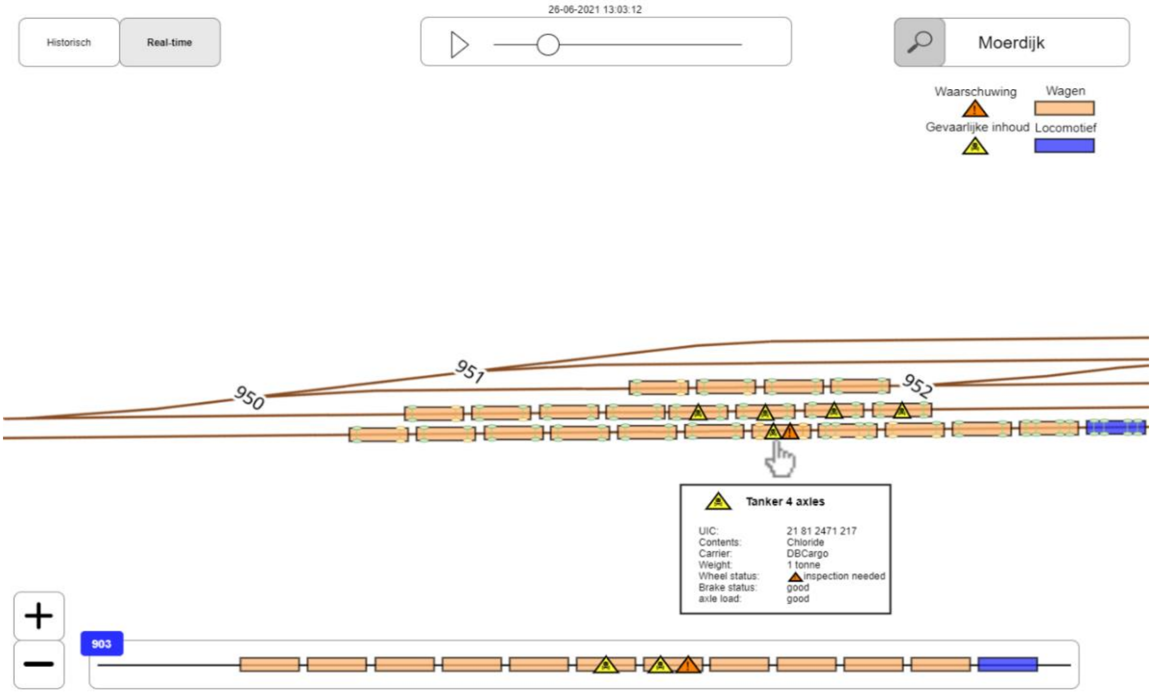


Figure 7.4: Digital twin GUI mock-up

same data set. When the user wants to look back in time, there needs to be some play button like a media player to view the turn of events. The map needs to show wagon-level information and should incorporate the possibility to view a set of wagons as the composition of a train. This could be done by showing the train composition at the bottom of the screen. For the need of asset health, we provided colour-coded wheels and axles, giving the possibility for quick hotbox detection, unround wheels or any other dangers. If there is an issue the system should warn the users, this is done through an orange triangle. Another aspect the system should make visible very clearly is when a wagon contains hazardous materials, which is done through the yellow triangles.

7.5 The requirements

We have designed several different aspects of the digital twin: the high-level architecture, the minimal requirement relational database scheme, the need for an event-sourcing alike architecture and a GUI to visualise the data in order to gain the needed knowledge. We can group the minimum requirements respectively.

7.5.1 Overview of requirements

The entire list of the requirements is given in Table 7.3, classified on the requirement level. The table shows a simple list of the four main requirements, and when relevant the constructing sub-requirements are listed below the respective requirement.

7.5.2 High-level architecture

The requirements in the high-level architecture are the blueprint of the system to have the basic foundations in the architecture. The overall requirement could be defined as:

R1. *The system shall support a 6-layered, modular architecture.*

Requirement	Requirement level
R1. The system shall support a 6-layered, modular architecture.	Domain
R1.1. The system shall allow gradual innovation through the low-effort extension of • The number of sensors connected • The number of RESTful sources	Product
R1.2. The system shall incorporate the REST and OPC principles	Product
R1.3. The system shall incorporate the MVC principle	Product
R1.4. The system shall implement an independent, automated wagon tracking and identification	Product
R1.5. The system shall implement an independent, automated asset health detection	Product
R1.6. The system shall have a log-in function	Product
R2. The system shall support a relational database scheme.	Domain
R2.1. The system shall incorporate compositions composed of one or more units	Product
R2.2. The system shall incorporate wagons and locomotives as units	Product
R2.3. The system shall have the option to transform a composition into a train	Product
R2.4. The system shall support the storing of events	Product
R2.5. The system shall never incorporate commercially sensitive data	Product
R2.6. The system shall incorporate the specific fields as given in *Figure 7.2*	Design
R3. The system shall support an event-driven architecture.	Domain
R3.1. The system shall check the memory for new sensor-data	Product
R3.2. The system shall generate events from incoming sensor data	Product
R3.3. The system shall incorporate a snapshot system for the database	Product
R3.4. The system shall apply events to the snapshot nearest to a user given time-interval	Product
R4. The system shall implement a GUI.	Product
R4.1. The system shall have a 2D geographically correct map of the rail network	Design
R4.2. The system shall have a button for a real-time mode	Design
R4.3. The system shall have a button for a historical mode with a time-interval selection	Design
R4.4. The system shall display individual wagons on the rails	Design
R4.5. The system shall display multiple wagons as one composition/train on the rails	Design
R4.6. The system shall display the movement of wagons and compositions on the rails	Design
R4.7. The system shall have an option to play or pause a certain timeframe	Design
R4.8. The system shall display asset health of the individual wagons	Design
R4.9. The system shall display warnings for risks and hazardous materials	Design
R4.10. The system should display wagon information of a selected wagon	Design
R4.11. The system should display train/composition information of a selected composition	Design

Table 7.3: Overview of all system requirements

Since it is not feasible to get the wagon position of Carriers (Section 6.1.4) and neither to place GPS devices on the wagons of the Carriers (Section 7.3.1), the wagon identification in the field should come from sources independent from the wagons' owners. The most common sensors for achieving this are Axle counters, smart cameras, track switch sensors, short circuit sensors. Furthermore, having the same reasoning for independence, there should also be some sensors incorporated for enabling asset health detection. This requirement stems from the wish to incorporate a risk-driven approach regarding hazardous rail freight (Table 7.1, goal 2). This leads to the last two sub-requirements:

R1.1 The system shall implement an independent, automated wagon tracking and identification

R1.2 The system shall implement an independent, automated asset health detection

And finally, there is a requirement that is of importance but could not find its way to be displayed in any model: security. This one is important to mention due to the liability concerns:

R1.3 The system shall have a log-in function

7.5.3 Object relationships

It is determined what data is needed for the stakeholders to achieve their goals, which is defined in Section 7.4.4. There is not much to add to the provided figure, and the section could be made into a requirement:

R2. The system shall support a relational database scheme.

The database scheme as shown in Figure 7.2 can be described by the typical "unit" and "composition" objects:

R2.1 The system shall incorporate compositions composed of one or more units

R2.2 The system shall incorporate wagons and locomotives as units

R2.3 The system shall have the option to transform a composition into a train

Then there is the need for the database to support the event-driven architecture, for which events that took place have to be stored:

R2.4 The system shall support the storing of events

One additional sub-requirement that should be added due to reasons given in Sections 6.1.4 and 6.1.6:

R2.5 The system shall never incorporate commercially sensitive data

And finally, we have shown many fields that are believed to be of use to all stakeholders, and these are incorporated into the model. These are requirements on the design level:

R2.6 The system shall incorporate the specific fields as given in *Figure 7.2*

7.5.4 Event-sourcing

The need for an architecture that supports asymmetrical event registration, such as the event-sourcing mechanism as described in Section 7.4.5.

R3. The system shall support an event-driven architecture.

This requirement should be interpreted as follows: the system is dependent on a multitude of sensors that generate data asynchronously and not bilaterally (Section 7.4.5). Therefore there should be an architecture that supports this observation, and in this thesis, the event-sourcing mechanism of Figure 7.3 was suggested. In order to find out whether new data has been inbound, the controller requests updates of the data memory:

R3.1 The system shall check the memory for new sensor-data

The incoming data will be transformed into events, that will be stored in the database. When a user requests a specific point in time, the controller selects the nearest database snapshot to which all events are applied until the requested state has been reached. This is incorporated in the following sub-requirements:

R3.2 The system shall generate events from incoming sensor data
R3.3 The system shall incorporate a snapshot system for the database
R3.4 The system shall apply events to the snapshot nearest to a user given time-interval

7.5.5 Graphical User interface

It is hypothesized that the GUI could provide knowledge to the associated stakeholders from Table 7.2. The requirements list in this section is more extensive than of the previous sections, as the GUI requirements are elicited on a design-level (i.e. more detailed functionalities). The product-level requirement in this section is:

R4. The system shall implement a GUI.

The needed functionalities are listed below, the explanation of why how these requirements came about is given in Table 7.2: Stakeholder goal GUI assessment. All of these requirements describe the GUI as shown in Figure 7.4.

R4.1 The system shall have a 2D geographically correct map of the rail network
R4.2 The system shall have a button for a real-time mode
R4.3 The system shall have a button for a historical mode with a time-interval selection

R4.4 The system shall display individual wagons on the rails
R4.5 The system shall display multiple wagons as one composition/train on the rails
R4.6 The system shall display the movement of wagons and compositions on the rails
R4.7 The system shall have an option to play or pause a certain timeframe
R4.8 The system shall display asset health of the individual wagons
R4.9 The system shall display warnings for risks and hazardous materials
R4.10 The system should display wagon information of a selected wagon
R4.11 The system should display train/composition information of a selected composition

7.6 GUI requirements implementation

This paragraph is mostly concerned with the validation of R4 and its sub-requirements (Table 7.3). It should be noted that the graphical interpretations of the corresponding requirements are on a creative basis: every developer could give it their visual interpretation, but the essence should stick to the theory. Therefore, the visualizations of this artifact are neither fixed nor determined, but they should reflect the core of the functionalities. This does not only concern the graphical characteristics of the artifact, but also the architecture could be implemented in many ways to support the same set of requirements. Therefore, also the design choices of the treatment with respect to the architecture are not considered to be the best solution for the actual treatment design but are considered as the best choice to effectively and efficiently test the requirements document.

Our artifact should support the planned single-case mechanism experiments, which have the need for simulated environments. The validation design will be explained in Section 8.1. The artifact will be called treatment as is common in design validation and should be able to support some kind of simulation to test the separate components of the requirements. This simulation should reflect real-world examples of the use-cases of the artifact. More on these simulations is in Section 8.1.

7.6.1 Tools

There are many backbone components to be implemented, and for all of these different components, many different tools or frameworks exist. Architectural components in an MVC system are, among others: a database system, a framework for the programmable logic of the controllers, a framework for the views. Tools selected for this research have been completely personal for which there is no objective reasoning: one could use its preferred environment.

To start, a framework had to be chosen for controlling inputs and outputs. A selection of three was made: the game-design application Unity, the Java programming language or implementing a web service (using TypeScript). Where Unity is focused on creating graphic scenes this piece of software would be perfect for creating beautiful scenic digital twins. In Java, programmable logic is the focus and libraries are existent to a full extent. Then there is the web service, an all-rounder that needs some creativity for such graphical representations, but web services are very accessible to a wide range of developers and users (one server is needed to feed all users, instead of hosting the app on all user's systems). The major downside of both Unity and Java, in this case, is that the view's size should be predefined: only a specific part of a map can be drawn as borders are predetermined. Besides these downsides, the researcher has the most experience in web services and therefore building a web service was chosen before the other options. Furthermore, OpenStreetMap has a railway spin-off OpenRailwayMap which is open source and freely accessible to implement in any web service as long as the developer complies with the open-source license agreements.

Then there is the choice of web-service frameworks. Again this is a personal matter, but Angular has been chosen with TypeScript as the dynamic programming. This makes that an extra server for the programmable logic can be skipped, as all scripts could be run on the Client-side. This allows for dynamic programming, such as moving a train without an overload of backbone calls. For the view, HTML is used, with the Angular and Google Material libraries for styling and templating. Then the missing component is the database. While we could implement a full database server to connect to our controllers, this is found to be overdone. The most time-efficient would be to implement a JSON-server that simulates a database. The only data that will be saved are the train positions in the simulations (i.e. scenarios).

To summarize, the following components have been used:

- NodeJS (to support npm)
- NPM (to install angular and all libraries)
  - Angular (with TypeScript and HTML)
  - JSON-server
  - OpenLayers (to support OpenStreetMap)
  - Material UI
- OpenStreetMap
  - OpenRailWayMap (is an OpenStreetMap add-on)

7.6.2 Twin interface

As mentioned, the interface is based on the interfaces provided by OpenStreetMap and OpenRailwayMap. Then, using HTML and TypeScript some wagons are drawn into the picture, as seen in Figure 7.5. The design aims to mimic some kind of media player that also features a real-time view. So on the one hand it is possible to select a timeframe in history and to play or fast forward through all wagon movements within that timeframe. And on the other hand, it is possible to view all current wagon movements. In order to restrain information overload to the user, the displayed information is based on the zoom level of the user. At the current zoom in Figure 7.5 the train level is taken, in which information about the full train is more important: number of wagons, length, movement and positioning, and so on. The only wagon-level information shown in this level is concerning the warnings: which wagon contains hazardous materials, and which wagon might need some further inspection before passing through. Once the user clicks on a wagon, at this zoom level the full composition is selected and the corresponding data is shown.

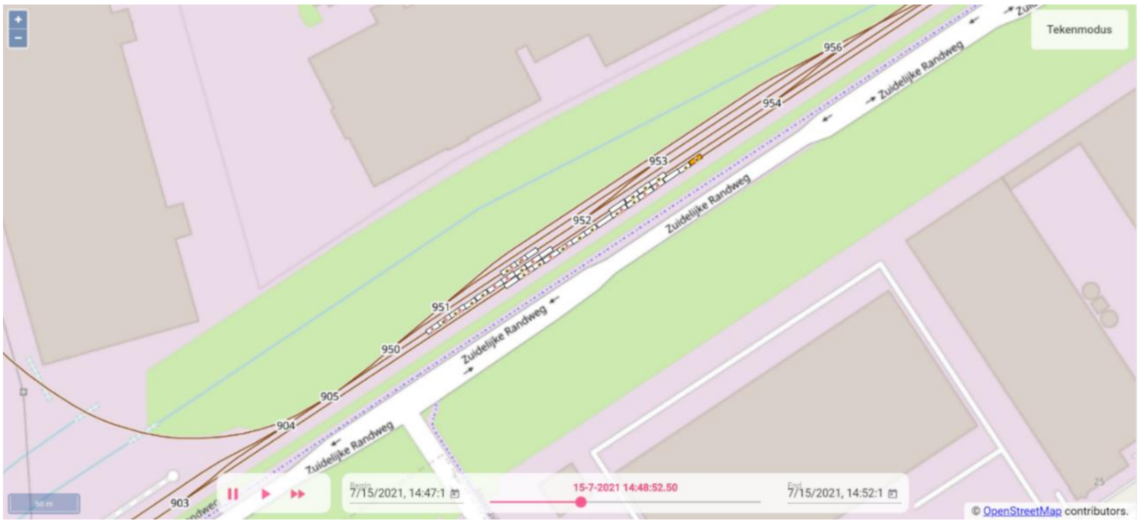


Figure 7.5: GUI interface zoomed-out

When the zoom level is increased (i.e. zoomed-in) to a certain scale, the information becomes wagon-focussed, as depicted in Figure 7.6. In this zoom level, the user is able to directly see all of the states of the wheels, brakes and axles through colour-coding. This should give the user enough information to detect the seriousness of a hotboxing wagon containing flammable substances. Clicking any wagon in this view shows all wagon-specifics, such as weight, size or contents.

Figure 7.6: GUI interface zoomed-in

7.6.3 Plotting wagons

All additional visualisations are all plain HTML fed by data through Angular. Wagons are JSON objects containing all necessary information. In terms of rendering a wagon, this needed information is the length, width, whether the wagon is a locomotive, whether the contents are hazardous and whether the axle-/brake-/wheelstates are in pristine condition. All there is to a wagon in this application is an HTML “div” object mapped onto the map through an overlay as provided by the OpenLayers library. A div has a size that is relative to the scale of the map: if the scale indicates a line for 20 meters, a wagon of 10 meters is half the size of that line. Then in this div, the properties of the wagons are plotted: some imaging is used for alerts or nested div elements for the axles and the wheels. As soon as the map gets dragged or zoomed an event is triggered to replot all the wagons on that specific timeframe with the new sizes and positions.

Displaying the wagons as plain HTML div blocks showed to be a challenge. OpenRailwayMap provides its users with static image tiles as the map. All maps, such as google maps, make use of tiles in which the rendered map shows tiles only in the current view: imagine wanting to see the map of the Netherlands, in this case only the tiles containing the Netherlands instead of all of the world are shown to reduce memory usage and make sure the tool works seamlessly. Besides positioning there is also zoom, in which tiles have a certain resolution. The tile sizes are fixed, but the information in them is based on the zoom level. A zoomed-in map of Enschede would show me in the same resolution more details than a map of all of the Netherlands. The biggest issue that arose here, is the fact that these tiles are images, so the big question is: how to orientate a rectangular block onto the slope of the railway? The OpenRailwayMap makes use of 8 different colours for rails, and therefore image processing seemed to do the trick. In Figure 7.7 this algorithm is shown: each tile has a fixed size of 512 pixels, and each tile has a range of coordinates. The coordinate of the wagon is known, and thus the centre pixel of the wagon could be computed. From this pixel, we create a 10 by 10 colour matrix of the surrounding pixels, in which a 0 indicates not the right colour of the railway and a 1 represents a pixel of the railway. The slope could be calculated by dydx and the inverse sin provides the slope angle to rotate the div block.

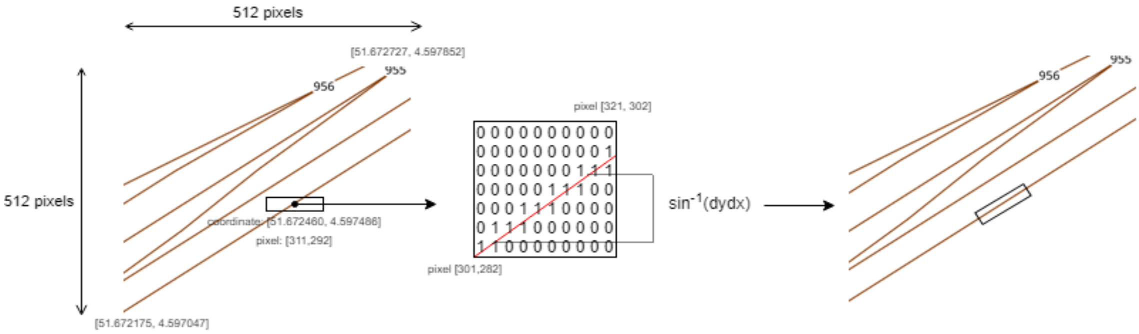


Figure 7.7: Image processing algorithm for wagon orientation

For positioning the wagons there are *wagonposition* objects: JSON objects as shown in Figure 7.8. In a production system, this would be very inefficient, and therefore this solution would be incompatible with any real-world artifact. The system works based on the “play” feature of the interface. The play feature takes the current timestamp and adds half a second and then waits for half a second and so on. Therefore, this timestamp is leading, and all generated *wagonpositions* should comply by having half or full seconds as timestamps. Conclusive, the play feature drives the current timestamp, and each half-second all *wagonpositions* are filtered on that specific timestamp.


```
{
  "id": 90622278529,
  "wagonId": 671203885,
  "pos": [
    512323.97119033354,
    6741451.083951083
  ],
  "timestamp": "2021-07-15T12:49:34.000Z",
  "rotation": 22.373015610004096
}
```


Figure 7.8: JSON object of a wagon position

7.6.4 Drawing scenarios

There is a final challenge to mention here: generating data. Generating data should be done in a way that is compatible with this tailored system, and the most efficient way seemed to be implementing a “drawing table” (tekentafel) feature. This feature allows the researcher to draw scenarios on the map. By clicking at points in the map, the researcher could draw a route for the train (or wagon) to follow (Figure 7.9). When the specific route for the desired wagon composition is drawn, the researcher could set the scenario for the composition, such as speed (or if stationary the amount of minutes it is stationary), the number of wagons, whether the composition has a locomotive and so on (shown in Figure 7.10). Then the script runs in order to generate random data suiting the researchers’ provided parameters. Based on the speed and route the script calculates at each half-second where each of the wagons is positioned. Measuring all distances is done by a reused script that takes the two points as coordinates and calculates the distances respecting the Earth’s curvature. When the script made the calculations, the scenarios are printed to files in JSON formats and could be used to be combined with other scenarios to implement a larger demo.

Figure 7.9: Drawing a train path

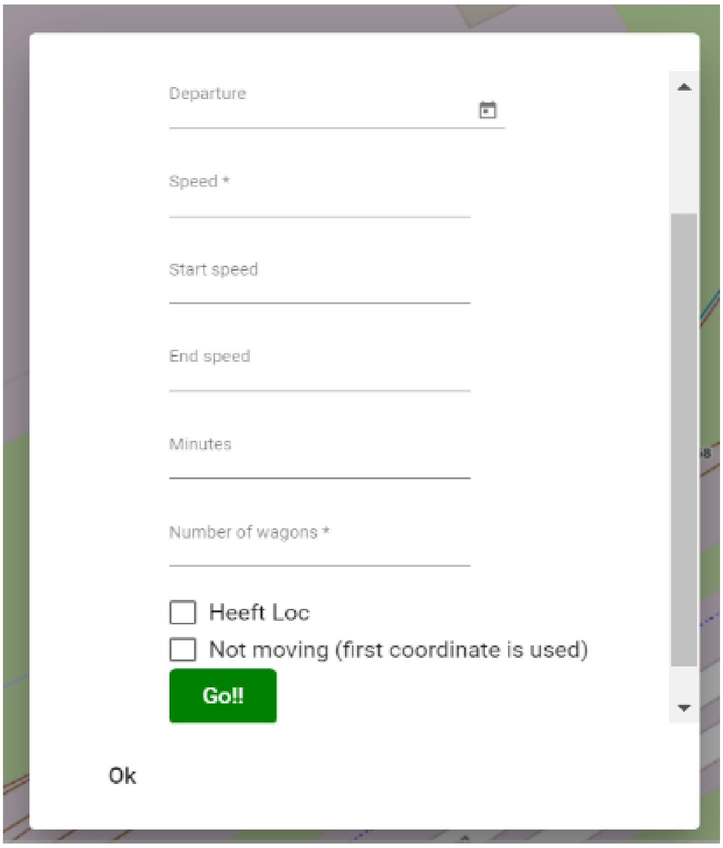


Figure 7.10: Train path parameters

8 Validation

TREATMENT VALIDATION

This chapter is the final phase of the design research cycle. In Figure 3.1 (page 27) on the right-hand side is the design cycle that was followed to the point of this chapter. The arrow in the figure shows that the empirical cycle is related to the treatment validation phase. The phases of the empirical cycle show the guidelines of the evaluation research conducted in this chapter. The conclusions will be provided in Chapter 9, in which also the validity threats to the research will be assessed.

8.1 The evaluation research design

First, we give an overview of the user evaluation research methodologies and instrumentation for evaluating the specific requirements in Table 8.1. Note: the instruments relate to the given methodology, but are not dedicated to a specific requirement. The following sections explain how the different methodologies are executed, introduce the reader to the expert panels and describe how data will be generated and collected.

Methodology	Instrumentation	Requirement
Expert opinion	Presentation (Appendix H), Microsoft Teams, Microsoft Office	R1. The system shall support a 6-layered, secure, modular architecture.
		R2. The system shall support a relational database scheme.
		R3. The system shall support an event-driven architecture.
Single-case mechanism experiment	Implemented treatment (Section 7.6), Microsoft Teams, Microsoft Office	R2. The system shall support a relational database scheme.
		R4. The system shall implement a GUI.

Table 8.1: Requirements by Methodology and Instruments

8.1.1 Evaluation methods

As described in Section 4.6, this evaluation phase consists of two separate evaluation studies employing two separate evaluation methods: first, an *expert opinion method* to evaluate the architectural aspects of the design and then a *'single-case mechanism alike method'* (as per Wieringa, 2014) used in the structure of scenario analysis. The overall evaluation goal of both studies in this chapter is to verify whether the designs indeed are a solution to the problem as described by the design problem: transparency issues that form implications to capacity management in freight train service and risk assessments in case of emergencies on rails. Furthermore, the system should be feasible and unambiguous (i.e. a logical solution), and it should describe the bare minimum viable product (i.e. no requirement should be missed nor superfluous).

As a major task in this evaluation research design, the simulations in which the single-case mechanism experiments should take place were developed. These experiments are to show whether the design-level requirements enables the users to gain the advantages of the digital twins' architecture (the specific fields of the data model and the GUI design). The expert panel that has been selected is composed of people knowledgeable about the specific user groups (the four stakeholders of Table 7.2). These experts were either the stakeholders themselves or experts with a tight connection with these stakeholders and knowledgeable about the stakeholders' interests.

Next to the scenario analysis, the expert opinion research is conducted for evaluating the architectural choices of the design, more on the product-level requirements (requirements 1, 2 and 3 of Table 7.3). The data model of Figure 7.2 (requirement 2) has some overlap: in the expert opinion it was evaluated whether the architectural choices such as the relationships make sense, and in the single-case mechanism experiment it was evaluated whether the fields were selected properly (the design-level requirements). The overview of requirements versus methodology is given in Table 8. In this table is also an overview provided for the used instrument for the requirements.

As in our evaluation research concerning the requirements, there is no hard data to "prove" the requirements (such as seconds faster, or per cent decreased costs), most of our evaluations are based on opinions, and therefore another set of interviews were used in this phase of our research.

8.1.2 Expert panels

The pseudo-randomly selected participants in both evaluation studies are displayed in Table 8.2. In the contact network of the company, the subjects were approached coincidentally: no pre-screening was done of the subjects (other than function and relevance to the case), subjects were approached were just available at the time. We have selected two expert groups for this validation chapter: one for the expert opinion, and one for the single-case mechanism experiments.

For the architectural evaluation of our concept design, some IT architects that have been active in rail logistics for at least three years were selected. We note that there have been many alike projects that were focused on the implementation of software in rail logistics regarding wagon tracking and identification. Therefore we pseudo-randomly select a few engineers from known projects.

The group of experts selected for the single-case mechanism experiment consists of subjects with knowledge of the stakeholders involved in the scenarios, i.e. the stakeholders mentioned in Table 7.2.

8.1.3 Data collection and instruments in the two evaluation studies

For each of the two evaluation studies, a different instrument was used, as is shown in Table 8.1. The expert opinions study was done as follows: first, the author of the thesis gave a short briefing on how the systems are designed by using a PowerPoint presentation (Appendix H). In detail, every design choice was substantiated, to provide the subject with the full context. Some required background information was given, to make the participants understand the full problem scope. After a quick check whether the subject understands the substance, the designs of requirements were shown one by one. Each requirement was thoroughly discussed and questioned. For each of the sections in the evaluation, the subject was asked for his perceived level of knowledge about the specific sub-topic.

The single-case mechanism experiments were conducted using the developed treatment as described in Section 7.6. This treatment has the option to draw scenario's that reflect some probable real-life situations. These scenarios arere explained in detail in Section 8.1.4. Using the scenarios as a point of discussion, the subjects were asked whether the design is understandable to them and reflects their contexts and whether the design would be helpful in the depicted situations. It is of importance that the subjects understand the roles of the involved stakeholders.

Methodology	Selected object	Introduction
Expert opinion	IT Architect I (Appendix I)	This subject has had many experiences in software design, requirements engineering, solution engineering and business process modelling. The subject has knowledge about ArchiMate and UML, and with its history as an employee of the Network Operator and experience in rail freight, the subject could place the artifact in the right problem scope.
	IT Architect II (Appendix J)	Where IT Architect I has a more business focussed approach, this IT architect has a very technical nature. The subject has studied computer science and graduated before the 2000s, indicating that there is a history of experience. The subject had worked for over 20 years in the same company and has a few years of experience in IT for rail freight, such as the WLIS application.
	IT Architect III (Appendix K)	This subject has worked for 6 years in the rail freight section of a consultancy agency as a software engineer. The subject is somewhat younger, and therefore has a good understanding of modern innovations in IT. He has worked on systems as TROTS, TOON and WLIS.
Single-case mechanism experiment	Capacity management (Appendix L)	This subject is employed by the Network Operator and is responsible for allocating capacity to all Clients. The tasks of this subject are all related to rail freight, and therefore the subject has knowledge about the proper context. The subject is also closely related to Traffic Control and has some interesting remarks on their behalf. Its knowledge is sufficient to give an evaluation of the requirements belonging to goals 1 and 3 of Table 7.2.
	Consultant IV (Appendix M)	This subject is a consultant with a history of projects in the field of rail freight with a focus on capacity management. This subject has played a major role in the Botlek PoC (Section 7.3.2), which had a focus on capacity management. Its knowledge exceeds beyond capacity management in the sense that there was an understanding of the tasks somebody of Traffic Control has. Therefore, just as subject “Capacity Management”, this subject also could help in validating goals 1 and 3 of Table 7.2 and their corresponding requirements.
	Consultant V (Appendix N)	This subject is also a consultant with a good understanding of IT projects in rail freight. Its experiences in the field have lately been focussed on hazardous materials on emplacements, with a knowledge of how Incident Management and the Emergency Services respond to unforeseen events. One example of its experience: the subject is an architect for WLIS. This makes the subject perfectly suitable for judging the mock-up. The focus with this expert will be on incident management, thus goals 2 and 4 of Table 7.2.

Table 8.2: The expert panels by methodology

Both evaluation studies are executed by means of interviews, the difference in method lies in the used supporting instrument. In Figure 3.11 the chronological order of inferencing is shown. First, there is a need for descriptive inferencing, to which abductive inferencing is applied (Wieringa, 2014). The findings of the abductive inferences are used for generalizing through analogic inferencing. First, the interviews will be performed through Microsoft Teams, with the recording function enabled. These recorded sessions will be summarized using Microsoft Word, in which a document is composed with only relevant and noteworthy quotes. Then, the raw interview data was shaped into useful and understandable data blocks: this is descriptive inferencing. Descriptive inferencing is done by documenting the recorded interviews as summarisations. These summarisations are tagged with notes, that conclude from what is being said (Appendices I, J, K, L, M and N). In these appendices, the black text indicates what subjects have said, and the text in red depicts the tags as part of the descriptive inferencing.

After the descriptive inferencing, the observations that led from the data processing were used to explain some observations. Observations in this study can be done through causality and rationality. This phase is abductive inferencing and is described in the “results” section (Section 8.2).

8.1.4 Scenarios

The treatment that will be used for the single-case mechanism methodology has already been specified in Section 7.6. The scenarios are still to be developed at this point though, this will be covered in these sections.

8.1.4.1 <u>TRS allocation (stakeholder goals 1 and 3)</u>

Ideally, the terms NCBG nor CBG are not needed anymore: no matter where any wagon or train is positioned, traffic control or process management would always know its location and whereabouts. This would be the major benefit to short-term planning of trains: the possibility for ad-hoc movements would be achieved and the hassles around TRS reservations would be no more. The latter is the important aspect displayed in this scenario.

It seemed that around TRSs a lot of analyses are performed for the optimization of their space and timeslot allocation. The former regards the way TRS areas are designated: the Network Operator is putting effort into dividing the spaces in NCBGs as optimal as possible to serve as many Carriers as possible. However, often there is only a single track to access several industrial areas. In this case, the NCBG factor plays a negative role: let us look at the scenario in the Botlek area presented in Figure 8.1.

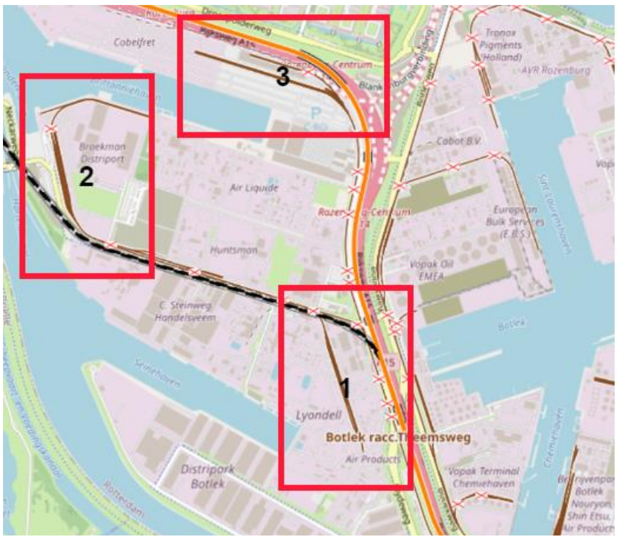


Figure 8.1: Botlek TRS allocation

In the figure the TRS allocation of Botlek is displayed, these are indicated by the numbers 1, 2 and 3. If either TRS 1 or TRS 2 are reserved, then there is no problem for any train to enter TRS 3. However, things get interesting when TRS 1 is occupied and another Carrier needs to access TRS 2. These TRSs share the same single track entrance, and therefore no train can access TRS 2 as there is no vision to ensure that the track in the NCBG is clear. Along the same line, no train could leave TRS 2 as long as TRS 1 is occupied.

This is caused by the matter of non-centrally serviced areas, there is just a lack of visibility to ensure a safe passing by of a train to TRS 2. That is what our scenario depicts: we plotted the scenario of a train loading in TRS 1, and of another train waiting to enter TRS 2. In Figure 8.3, the highlighted train on the right-hand side (from which its composition is shown in the upper information bar) wants to pass by towards TRS 2 while the train on the left is currently loading at TRS 1. Nowadays this scenario would be impossible, as it is unknown where the train on TRS 1 is positioned: it might as well be on the single track entrance risking a potential collision. But now imagine traffic control seeing the scenario shown below: they could signal the right train to pass by before the train on the left leaves. This is shown in the bottom hand of Figure 8.2. Although hardly visible due to the zoom level, on the left we see the same highlighted train as the one on the right in the upper situation. The other train could be seen leaving the area on the right.

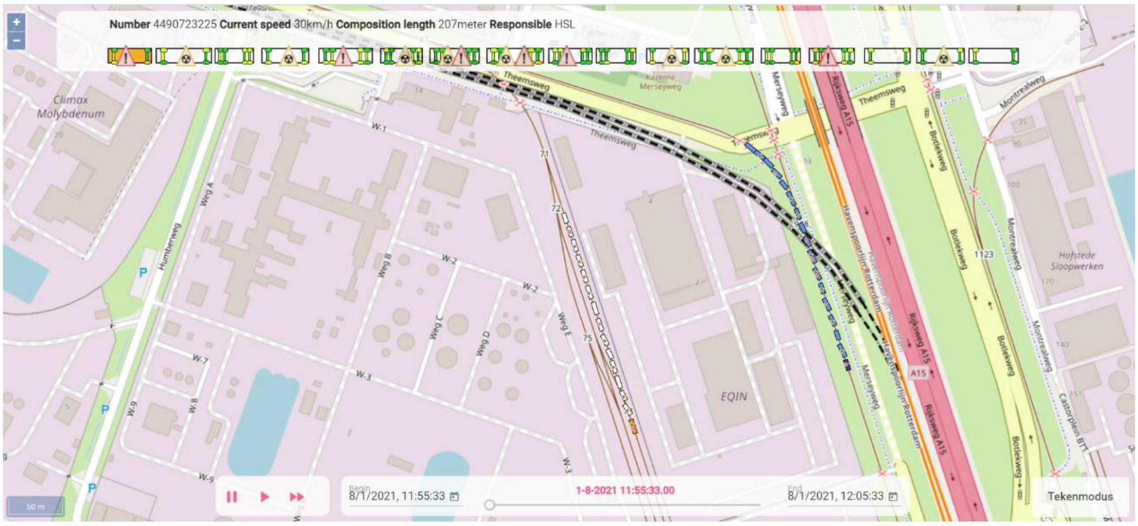


Figure 8.3: Train awaiting the signal to enter TRS 1 and 2

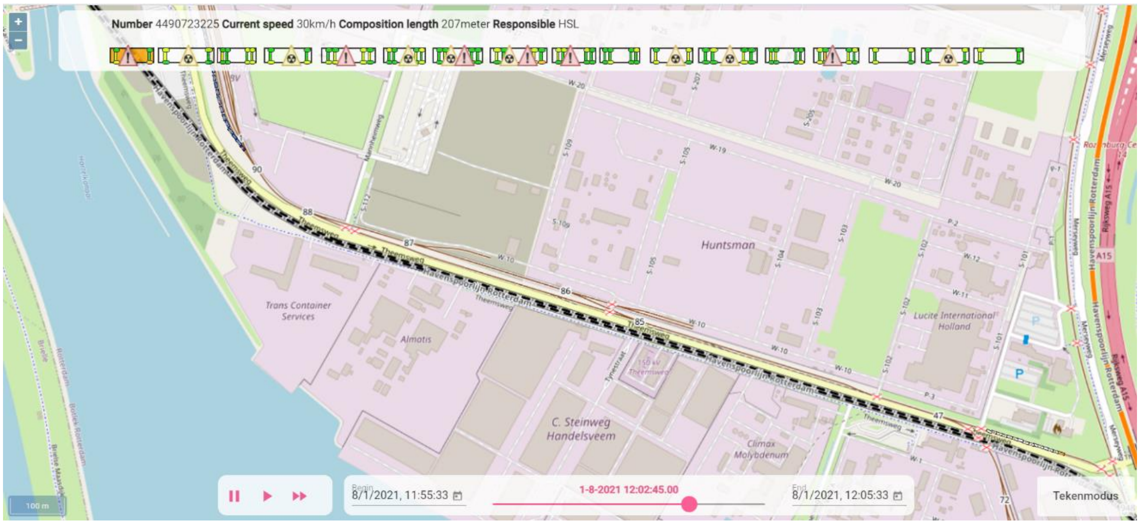


Figure 8.2: Train passing by to TRS 2

Then there is also the need to mention long-term planning and allocation of TRSs. The generation of the new so-called “basisuurpatronen” (basic hour patterns) already starts a year and a half before the execution of the new planning. Most of the traffic flows are static routines, and therefore most of the allocation of capacity is done through generating these patterns. However, sometimes some Carriers have conflicting desires resulting in table consultations for finding solutions. Having this visual aid could improve understanding complex situations on the emplacements, and help to understand the shunting behaviour of Carriers. By acting as a Common Operational Picture, the system could improve TRS allocation by increasing the shunting efficiency of the Carriers.

8.1.4.2 Incident Management (stakeholder goals 2 and 4)

Full visibility on a geographically correct data representation allows for several improvements to be made in responding to risks and incidents. Imagine the scenario on the Emplacement in Figure 8.4. The scenario displays two occupied tracks on emplacement Moerdijk, in which both wagon compositions contain flammable contents. With this updated view, every involved stakeholder would be able to detect quickly the risk level of exploding surrounding wagons. Besides, using a real-life presentation of the location makes that the services can more quickly find a wagon that is reported. In the current situation, the responding services have to count wagons on the tracks in order to create an understanding of the situation. The technology of having real-time information at hand could have interesting potential innovations, such as Augmented Reality solutions to provide services with a heads-up display with ad-hoc wagon information.

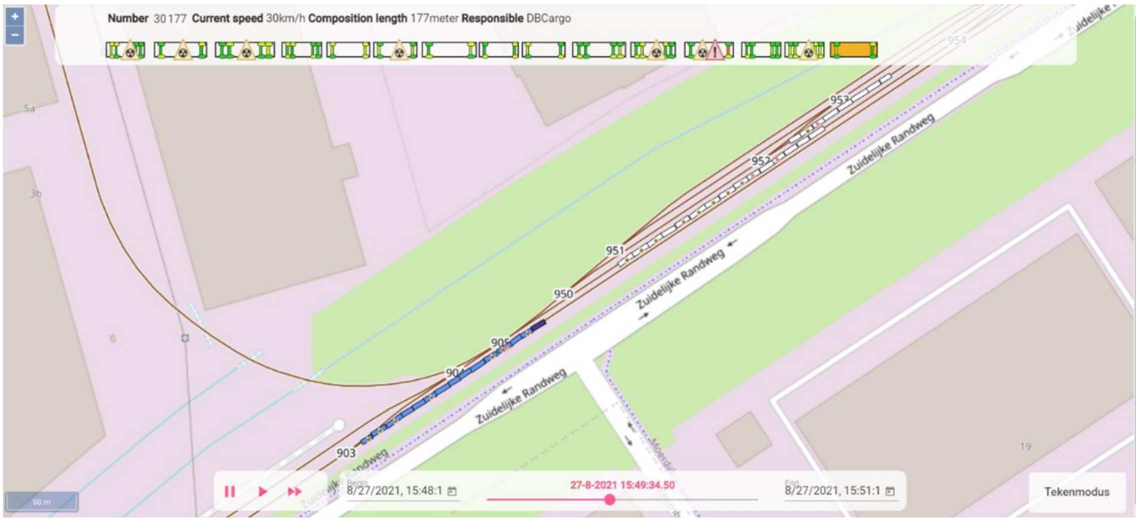


Figure 8.4: An emplacement with installed wagons containing hazardous materials

Then the final scenario, as depicted in Figure 8.5. Imagine the emplacement being in a densely populated area. There is a train entering the emplacement on the left-hand side, and the train is carrying flammable contents and should be handled with care. A glance at the system will tell you that this train has a hotboxing issue at the front axle, and therefore, shouldn’t be granted access to the emplacement to divert the risk of an incident with great consequences.

8.2 Results

This section provides the result of the executed research. The descriptive inferencing as done in the provided appendices will be transformed into conclusive statements (i.e. the abductive inferencing). In Table 8.3 an overview of the requirements is given with the extent of their acceptance. The full acceptance table of all requirements is found in Appendix O. The results of both validations will be explained in the following paragraphs.

Figure 8.5: Train with a warning sign passing by

8.2.1 Results from the expert opinion study

The expert panel was effective in providing feedback, both on the clarity of models (i.e. whether developers could understand them), and on the logic of the models (i.e. whether the choices made are valid). The coded interviews that have been the result of descriptive inferencing are shown in Appendices I, J and K. This section will be split into the 4 sections of the expert reviews: first the high-level architecture (Figure 7.1), then the object relationships (Figure 7.2), followed by the event-driven architecture (Figure 7.3) and finally some general observations will be written down.

R1. The system shall support a 6-layered, secure, modular architecture (Figure 7.1).
Mostly accepted. The architecture needs some tweaking, such as incorporating an event hub, but overall the solutions seem like the logical ones.
R2. The system shall support a relational database scheme (Figure 7.2).
Mostly accepted. The scheme was found completely logical. However, it was decided to discard the design-level requirements due to the project scope.
R3. The system shall support an event-driven architecture (Figure 7.3).
Partially accepted. The architecture as shown in Figure 7.3 was meant to reflect an event-driven mechanism, but in the end, was not. The overall idea seems right, but the fact that the controller actively asks for updates, rather than being triggered by events shows the opposite of event-driven. Having the same processes, but rather triggered by the event-hub makes the model acceptable.
R4. The system shall implement a GUI (Figure 7.4).
Rejected. This level of detail is too early for research, just as with requirement 2, all design-level requirements were discarded. It was mostly misinterpreted that all stakeholders could benefit from the same GUI. Even though the stakeholders all might benefit from the data and insights provided by the twin, they all have different desires from how the data is presented. Not always is the 2D geographically correct design the right one.

Table 8.3: Requirement acceptance as per the evaluations

8.2.1.1 High-level architecture (requirement 1)

All subjects were surprisingly positive about this architecture. The expert panel agreed on the logical order of the layered structure. On the modelling language, it was said that the “Digital twin” block was confusing (Appendix I notes 1 and 3) and that it was illogical to have the GUI and Controller as different types (Appendix I note 2).

Layer 3 has been actively been confirmed by the subjects, by incorporating examples of practice. There are currently several ongoing projects that make use of such an intermediate hub. The virtual train of Section 2.3.6 makes use of such an architecture (Appendix I note 4), many IoT based architectures have a similar setup (Appendix I note 7), and applications of the Network Operator have similar architectures (Appendix K note 1).

The virtual train example has also been used to confirm layers 4, 5 and 6 (Appendix I note 6 and 7). Overall, all the subjects agreed heavily with the modularity, this is a requirement that should be emphasized more. The MVC architecture has been agreed on due to this reasoning. Modularity was even said to be the main reason why the system would be feasible (Appendix J note 13).

The subjects agree that the information of third parties is needed, but REST might be too specific. Besides, REST is a mechanism that works differently compared to the event-like architectures of the applications of the Network Operator. This links to the lack of event-driven characteristics in the model. IT Architect II (Appendix J) and IT Architect III (Appendix K) very rightfully demanded changes in the current models, regarding event-sourcing. All subjects agreed with the requirement of having an event-driven architecture, but it was found that the models did not depict this. The model was lacking a “service bus”, or a “messaging queue”. In event-driven architectures, systems are subscribed to certain topics (these are certain events dispatched in the service bus) and act accordingly when topics are released from the queue. Therefore: layer 3 should add the events of the sensors in a general fashion to the service bus. The controller is subscribed to the service bus and applies mutations as soon as the service bus releases an event from the queue. As long as the controller is not fetching the updates, the topics are stored in the queue to make information asynchronous. These changes will be applied in the model (Section 9.1.2).

8.2.1.2 Object relationships (requirement 2)

The observations made here are quite short: all the subjects agreed with the model. The relationships are clear, the choices made are logical and the modal was not complicated. There was one observation made by the subjects: the locomotive entity is empty. This was known, and it is chosen to maintain the design as such: the model emphasized relationships, how the scheme is implemented is for the interpretation of the developers. Furthermore, it was found in all of the interviews that “event” was a very confusing word in the context (Appendix I note 16 and Appendix J note 6): “mutation” should be a more clear term.

One thoughtful observation that was made by the third interviewee (Appendix K): “A train number is unique for each day. So it could be the case that when a train departs at 23:00, it has the same train number as a train departing next morning at 7:00.” This indicates that the “train” field should incorporate a date.

8.2.1.3 Event-driven architecture

This was a heavily criticized model but will be accepted with some changes. Where Figure 7.1 got the criticism of not incorporating event-driven mechanisms, this model all the more. The model should depict an activity diagram of event-sourcing, but it is in no way found to be event-driven, as the controller is taking initiative by requesting updates (Appendix J notes 7, 8, 9 and 10 and Appendix K notes 9 and 10). The controller should not be looking for updates continuously, but rather the service bus should trigger updates. When the service bus puts a new update in the queue, the controller should read this and generate a “mutation” (what was an “event” before), and store the mutation in the database.

This is a major change, but after the first part, the subjects agreed on the system. It was just found to be too intricate, unnecessarily. The “event” naming provided a lot of confusion once again and will be renamed to “mutation” which captures the essence better. The snapshot idea was found very useful in downsizing processing time, and practice examples were given (Appendix K note 8).

8.2.1.4 General observations of the expert opinion phase

In general, all the subjects agreed with the requirements, but there were some alterations needed, mostly based on the event-driven mechanisms. The subjects have often emphasized the need for modularity, confirming the approach that was taken into the design phase by respecting the MVP and the Lean concepts. Furthermore, the designs were found feasible, both supported by the modularity and the real-life practice examples: similar systems have already been realized. Regarding the MVP: the subjects agreed with the level of abstraction. Neither too much was incorporated nor too few.

IT Architect III (Appendix K) had some fascinating statements. The first is regarding the feasibility: even though the subject thought of the system as feasible, it was necessary to point out the intricateness of layer 3. In other proof of concepts, it was found very difficult to standardize all different sensors by different vendors. This might need some extra attention.

It is worthwhile noting, however, that the expert panel identified an improvement point regarding scalability. While modularity is the leading concept in the thesis, the expert panel indicated that scalability has slipped. Imagine many sensors producing loads of data, it could happen that the queue growth faster than the controller processing messages. In this perspective, scalability was forwarded by the subject (Appendix K notes 4 and 12). Multi-threading and dockers were mentioned. The latter being a rather immature innovation shows lots of potential in horizontal scaling of processing power. Imagine being able to gradually execute multiple instances of an event processing function (or the controller), in this way you could divide the queue into several instances and execute the processing simultaneously.

8.2.2 Results from the single-case mechanism experiment study

This study brought an unexpected finding. The evaluation interviews gave different outcomes than as first hypothesized. The evaluations that are used in this section are provided in Appendices L, M and N. Overall the findings were as follows: the design-level requirements will be rejected (partly requirement 2 and requirement 4 fully), modularity of the system should have the primary focus and the main benefit of the twin would be to serve as a COP.

8.2.2.1 Data scheme

Some useful insights came about regarding requirement 2 (Table 8.3). First of all, the event table is not evaluated in this section, as it regards architectural design choices. The fields with regards to asset health have not been evaluated properly in the interviews, as the asset-health detection feature has been focussed on very little by the subjects. Only Consultant V has briefly outspoken its opinion regarding the alerts for health detection (Appendix N note 10), but in such an abstract way that requirement 1.2 can neither be rejected nor accepted. Asset health generation seems to be a feature that is too far sought and not ready to be incorporated in the MVP design. It should first get some more attention before incorporating it into the design.

Consultant V has emphatically mentioned the importance of the UIC and the responsible Carrier (Appendix N note 9), where the latter has been supported by Capacity Management (Appendix L note 19). Furthermore, Capacity management was outspoken about the incorporation of the remaining green fields (Appendix L note 19). The subject named these specific fields as “nice to have” for performing its tasks.

Consultant V rightfully pointed out the risks of adding a field “contents” (Appendix N note 8): showing contents disregards requirement 2.1 by neglecting privacy, provides implications for unloaded but uncleaned wagons, and it also adds to information overload, a term that has been the mainline of this interview. The contents field should be changed to what labels are applied to the wagon. This is a standardized field that directly shows the right information to the involved parties. Finally, also the “totalMass” field should be made more specific to the mass of the contents. The field as of now could give the implication that the mass of the wagon is also included. The latter is not useful, but the former is a requirement according to the subject (Appendix N note 6).

8.2.2.2 Graphical user interface

This section focuses on requirement 4 (Table 8.3). A major find in all interviews was that the required visualisation *differed* for each of the stakeholder groups. For example, to the Capacity Manager, it was found that a geographical representation of the infrastructure was of less value, but rather data visualisations would be needed in a dashboard-like manner (Among others: Appendix L notes 20 and 21). And even though Traffic Control does have its use for a map with train locations, to them it might be more useful to have the tracks schematically rather than geographically correct. For the Emergency Services the order of the wagons is especially important, and providing this information as a list might be faster and more intuitive than providing an actual map. These findings were backed up by the evaluation with Consultant IV, in which the subject had to be creative to find advantages of the system (Appendix M note 11). This subject had to make effort to summon the advantages of such a visualisation. Both evaluations, boldly put, lead to the rejection of requirement 4 and the enforcement of requirement 1.3: with all stakeholder groups needing different data and data presentations, the need for a modular system has become more important.

Adding to the modularity part: the requirement exceeds beyond adding different views. Both the Capacity Manager and Consultant V spoke of gradually adding new areas to the system. Investments made for the automatization of wagon tracking are too expensive to be made at once: sometimes a single PoC might exceed 75 million euros (Appendix L note 17). Also, Consultant IV made sure to mention the technology advancements: the subject believes that the twin is too early and innovation is still too advanced (Appendix M note 3).

Due to the rejection of requirement 4, of the different findings, only some noteworthy observations are discussed, as follows. Firstly, many sub-requirements could not be judged. Requirements 4.10, 4.11 and 4.12 are requirements that are necessary but are presented wrongly in the GUI. For instance: the wagon information is mandatory, but the way it is currently presented is too complicated and makes knowledge acquisition too slow for the users (Appendix N note 9 and 10). Furthermore, requirement 1.1 needs some extra attention: parties agreed to the demand for automated tracking and identification. But, liability is of concern. Carriers are blamed to be lacking in their shunting movement registration, and having a system to provide these automated registrations might only make them more careless (Appendix N note 4). To requirement 1.1 should be added that the WLIS data should be leading and that the automated identification should be utilized for the verification of supplied data by the Carriers.

8.3 Concluding remarks

8.3.1 Summary of the most important findings

This thesis did a two-step validation of the requirements: expert opinions on the architectural design choices, and single-case mechanism experiments on the design-level requirements. Below, we provide a brief summary of what is found. Overall, the validation phase led to the rejection of all design-level requirements: it was found that this level of detail is too early at this stage of the research. Furthermore, R3 was approved only partially: some changes are needed to get the requirement accepted. R1 and R2 were mostly accepted: all the subjects agreed with the designs, but some minor changes are in place.

The *expert opinion study* showed that the high-level design was rational and some real-life examples were given to back up the designs. There was one flaw in the high-level architecture: even though event-driven was propagated, this mechanism was not found in the architecture and should therefore be added accordingly. Moreover, R3 should have reflected an event-sourcing mechanism but in fact, displayed a continuous connection. The concept of event-sourcing was approved by all subjects, but the figure did not show this at all. The figure needs the right design alterations and could be accepted afterwards. All of the design alterations and the proposal of an updated design will be given in Section 9.1.2.

The *single-case mechanism experiment study* gave very different results from what we expected. It was thought that layer 6, the GUI, would be the most interesting aspect of the twin. However, it quickly turned out that each of the stakeholders required different views. Even though the GUI might be somewhat useful, it was found that all stakeholders would have liked to see different alterations. This goes hand-in-hand with the decisions made for the incorporated fields in the data model. All-in-all layers 1 to 5 were found to have the value at this point, and the design-level choices should come later. Therefore all design-level requirements of Table 7.3 were dropped.

8.3.2 Limitations of the validation research

We assess the threats to validity in the studies performed according to the guidelines of Wieringa (2014). First of all, we need to address the number of selected experts: both validation methodologies are composed of 3 experts. As this research makes its first evaluation of the design science exercise, having 3 experts in each study might be enough for drawing some indicative conclusions when we would only regard statements made by all of the experts. However, some of the experts made striking statements that the others had not thought of. These quotes were incorporated, but not backed up by any other expert. Therefore, to increase the validity of the study, the expert panel should be extended and re-run in the future.

Furthermore, the selected subjects for the single-case mechanism experiments included only one subject with the exact role of a stakeholder. The other experts did have a tight connection with the stakeholders, but they did not operate in the same environments as the stakeholders. Therefore, some issues might have been missed at this point. However, no acceptance came out of this study, and no false positives are the result of it.

And finally, some form of bias may come from the proposed designs. It was not asked how an expert would design a certain aspect, but rather a design was shown and the expert was asked whether he/she agreed. In the former, the expert would independently propose a design that could be compared to the proposed designs. However, this would be energy-consuming and certainly given the scope of this thesis, such an “internal validity” would not be feasible.

9 Conclusion

This is the final chapter of the thesis. All of the given information in previous chapters converge to the conclusion, in which the answers to the questions are provided. Then, the results are discussed on what limitations and assumptions shape the scope of the generalisation. Finally, recommendations on design and future research are given.

9.1 Final answers to the research questions and contributions

The main question of the thesis to answer is:

What constitutes a good digital twin design to solve transparency related issues in rail freight?

9.1.1 Problem investigation

To answer the main research question, it first had to be found out what the exact issues in rail freight entail. This is done by finding the answer to four sub-questions (SQ1, SQ2, SQ3 and SQ4: Section 1.4 Figure 1.4), of which the first question being:

SQ1. What communication flows characterize the rail logistics chain?

For answering this question, first, the involved actors in shipping freight on rails were determined:

- **The Client/Forwarder**: this is the owner of the cargo that needs its goods delivered from origin to destination
- **Rail Carrier**: This is the stakeholder that is responsible for shipping the Client's goods on rails. The Carriers have their own fleet and their own timetabling that is made in consultation with the Network Operator
- **Train Operator**: This is the employee that drives the train, often employed by the rail Carrier. The Train Operator is responsible for executing the timetabling of the rail Carrier.
- **Network Operator**: This party is responsible for maintaining the rail network and managing and allocating the capacity between the Carriers. The Network Operator makes sure that the traffic is according to the timetabling, and ensures safety on and near the railways.
- **Terminal Operator**: responsible for transshipping cargo. Incoming cargo is unloaded from trucks or ships and is loaded onto the trains. Or the other way around.
- **Emergency Service**: the national or regional service that provides assistance whenever an emergency arises. This stakeholder is of importance to ensure the safety as needed by the Network Operator.

The many flows of information between these six stakeholders are shown in Figure 4.1. Most flows of rail freight are fixed, and the communication sequence that occurs is repetitive. There is intensive communication between the Carriers and the Network Operator. Carriers have to request capacity which is allocated by the Network Operator. As soon as the Train Operators are executing the resulting timetable, active communication is maintained for signalling and keeping track of movements. In NCBG areas there is no automated tracking and these movements are diligently shared between the parties.

The Carrier is also involved with the processes of the Terminal Operator. The Terminal Operator requires instructions of the Carrier in order to perform the transshipment of goods. There was no communication between the Terminal Operator and any party other than the Carrier.

Finally, it was found that Emergency Services have an important role in the chain: even though there was not a defined active communication flow. However, due to the risk of transporting hazardous materials being incorporated, the Emergency Services are indispensable in the sequence diagram.

The operations that came forward as an answer to this question support the blueprinting of the IT landscape: the result of answering the following sub-question:

SQ2. What IT architecture supports the business needs in rail freight?

In literature and practice, many different types of software involved in the rail freight processes were found. Interviews with stakeholders in the field provided a lot of supporting evidence of similar pieces of software. All these different types of software were grouped into 12 different generalised classes (Section 5.1):

1. Consignment note systems
2. Emplacement reservation systems
3. Information management systems
4. Order to cash systems
5. Path coordination systems
6. Terminal operating systems
7. Train detection devices
8. Train dispatching systems
9. Train information systems
10. Train protection systems
11. Train timetabling system
12. Wagon information system

These applications are mapped in Table 5.1 to the operational flows that answered SQ1 (Figure 4.1). Through this table, and by modelling the landscape through a case study of Dutch rail freight, a general landscape was blueprinted in Figure 5.1. Again, where in SQ1 it came forward that the Network Operator is a central entity, the network came in the blueprint forward as the main digital initiator. The Network Operator seems to have such a crucial role, that it was deemed necessary to split the stakeholder into three roles: Traffic control, Capacity management and Incident management. Most applications are administrated by the central entity, with which the Carrier and Emergency Services also interact actively. An interesting observation was the silo in which the Terminal Operator exists, there are no active digital communication flows from and to this stakeholder. Furthermore, it was remarked that the wagon information systems and the train information systems were crucial for maintaining safety and increasing capacity.

The case study used for SQ1 and SQ2 lead to the disclosure of recurring problems in the field with regards to IT support. This is encompassed in the following question:

SQ3. Where is digitalization lacking concerning information needs?

The challenges are outlined by six domains:

1. **Transparency**: This was the most essential challenge of the domains. The Network Operator struggles with the transparency of where trains and wagons are positioned on the network. This is especially a problem in the NCBG areas, where there is no automated tracking system. Having inaccurate data sometimes provides incorrect images to Capacity Management and Incident Management.
2. **Data accuracy**: Transparency has the focus of the Network Operator by designing systems with which movements and train compositions could be registered manually. However, manual data generation is prone to errors, especially in a field in which margins are low and schedules are tight. It was found that almost 40% of the registered wagons were registered wrongfully.
3. **Utilization**: The previous challenge domains lead to issues with utilization. This is quite significant, bearing in mind that ambitious goals are set to double capacity by 2040. Lacking transparency and data accuracy make capacity management a complex task. TRS reservations are made too broad and congestions arise on the network. These issues are mostly in NCBG.
4. **The willingness to cooperate**: This is more of a cultural issue with a relationship with digitalisation. Innovations are expensive, and parties are not willing to make them if the investment does not provide direct value. The Carriers show no intention of sharing data with the central entity, and there is no collaborative culture between competitors. Innovation should be independent of the non-central parties.
5. **Automation**: This domain has a relation to transparency and data accuracy. The automation of processes is lacking when it comes to operational transparency. There are ambitious projects for self-driving trains, but the technological advancements lack in the surrounding fields. Before these advancements could be fully realised, first automated transparency needs to be realised.
6. **Liability**: The last domain is an indirect challenge that comes with innovation. With automation and transparency, there arise new weaknesses that need to be covered. For instance: when AI registered a wagon wrongly and that specific wagon is involved in a catastrophe, who is liable? Or imagine having too much transparency, and commercial data is known about installed wagons: criminals could get a literal treasure map. And in NCBG, theft is a serious issue.

The first three questions lead up to the problem domain that will be the focus of the digital twin:

SQ4. How could digital twins be a solution to information inadequacies in rail freight?

The information inadequacies mostly are regarding transparency or are a cause of the lack of it. This leads to the main challenges that need to be overcome first and foremost: ineffective capacity management and risk assessments. If a system would offer a better understanding of what happens on NCBG, issues regarding the transparency and the data accuracy domains could be solved. Although the emphasis is on NCBG, the existing systems supporting CBG could also be incorporated.

Digital twins are already existent in the field, but these are mostly on the infrastructure asset-level: predictive maintenance and product lifecycle management. There is one example of a process-level digital twin in Dutch train traffic (Section 2.3.6), but this does not regard the freight of cargo. The designated challenges in rail freight could be overcome by having a process-level digital twin in rail freight. This could entail a 2D map of wagon and train positions with additional composition information.

There should be taken a Lean and MVP perspective: innovations are expensive, technological advancements come gradually and current solutions should be acknowledged. Therefore, the system should be designed to serve a focused purpose of solving the core problem. No unnecessary implementations should be made: how the system advances should be assessed constructively. The current solution in the IT landscape that could be used are, among others, the wagon information systems, the train information systems and the train detection systems.

9.1.2 Design recommendation

Based on the outcomes of the problem investigation, a design was crafted and validated, as described in Chapter 8. The validation lead to the fine-tuned designs that will be presented in this section. Therefore, this section will consider SQ5 and SQ6 together as a final answer to the main research question. In line with the MVP perspective, only the most central stakeholder goals were considered, the stakeholder goals with direct relation to the issues as described in the design problem:

Improve transparency with an emphasis on NCBG areas by designing an MVP 2D digital twin that satisfies business requirements of chain parties and innovation growth prospects while maintaining data security in order to improve capacity management in freight train service and risk assessments in case of emergencies.

This leads to a design that was focused on the goals of:

1. Traffic control
2. Incident management
3. Capacity management
4. Emergency Services

The Redelinghuys (2020) layered digital twin architecture (Section 2.3.4) has been the basis of the high-level architecture (Figure 9.1). All six layers are incorporated, with the needed additions to stay in line with the lean perspective of gradually innovating. Modularity and scalability are two keywords that are shared among many experts in the field, the idea of lean has been embraced warmly. The system incorporates the MVC architecture: Models are defined in the database of layer 5, the Views could be connected in layer 6 and the Controller is the connecting element in layer 4. The MVC principle ensures modularity: different models and views could be connected without much effort. It was found that many different stakeholders had divergent desires and interpretations of the right data visualisation.

Furthermore, in order to use existing technology, there is a need for two types of connecting elements: RESTful APIs are needed to incorporate data from different sources. This is a standardisation that makes the controller modular in the number and range of sources that are implemented. Then, there is the event-hub that serves two purposes: to take care of the asynchronous nature of sensor data acquisition and to be able to dynamically connect new event-triggering systems. A mechanism that is regularly used in the landscape of the Network Operator. This makes the controller also modular in the number of subscribed topics (a topic is a subscribable event dispatcher), and thus in the number of connected systems. The event-hub enables event-sourcing: a mechanism that is found to be important to the architecture. Event-sourcing enables an asynchronous data flow and provides the possibility of horizontal scaling of processing power.

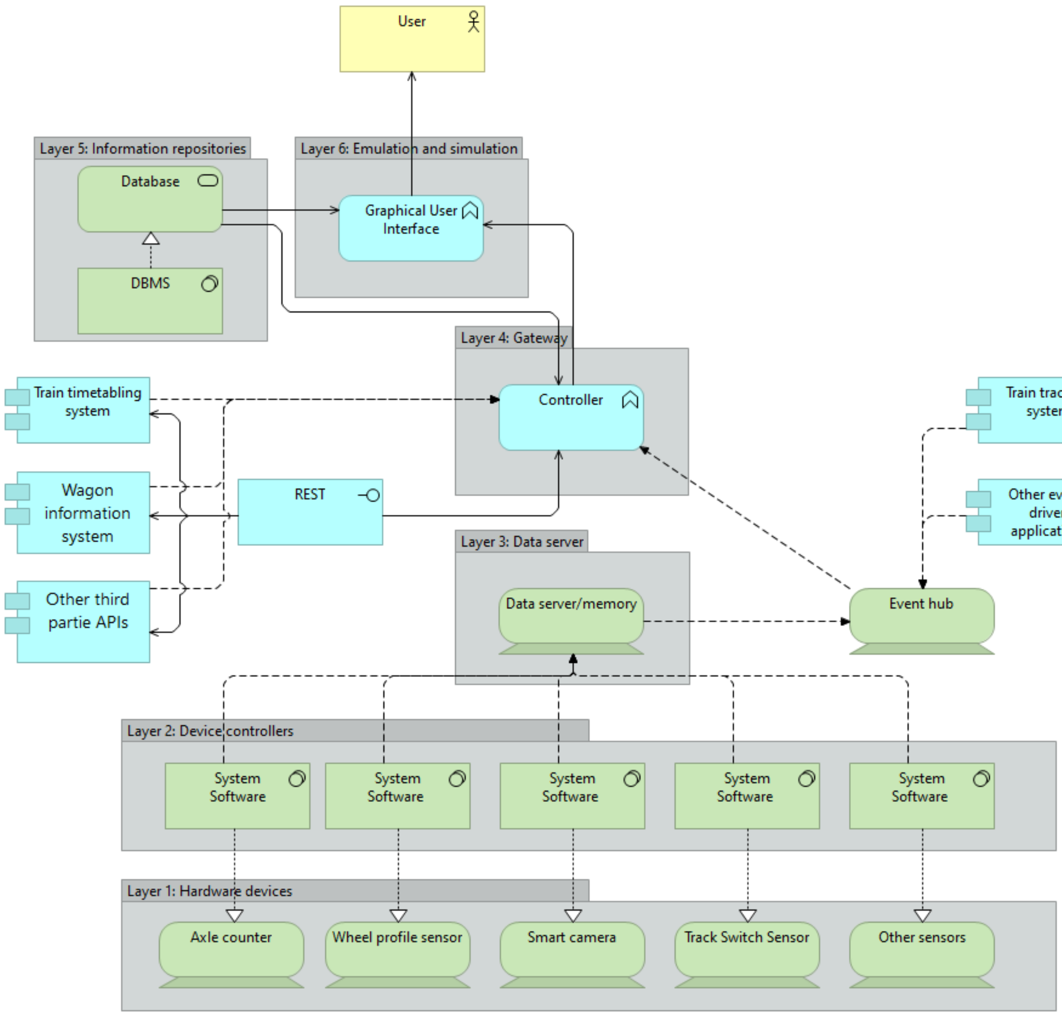


Figure 9.1: Validated high-level design

Onto the minimal required objects and their relationships that shape the model of the MVC (Figure 9.2). The objects are without fields, as the fields are in the design-level requirement domain that has been found outside of the scope of this thesis. The system should acquire data on the individual wagon level and the train level. However, there are a few important abstractions to make: trains are compositions of multiple cars with one or more locomotives. However, a composition could also entail a set of wagons installed on an emplacement, without a locomotive. That is why it was designed as such, that a train is a composition, but a composition is not necessarily a train. Furthermore: wagons and locomotives would have many similar fields. That is why abstraction is made here: a unit is either a wagon or a locomotive.

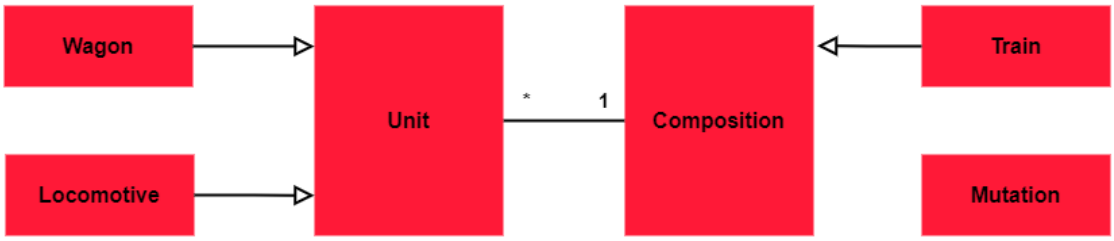


Figure 9.2: Validated object-relationship model

The final aspect of the architecture will be how the controller handles incoming events, as dispatched by the event hub. This “event-driven” architecture is vital for the twin. The twin should be able to playback events in a chronological order, which per definition gives the requirement of saving all proceedings in the database. Besides, when considering the number of sensors generating data at different frequencies, the need to buffer the asynchronous flow of data becomes evident. This leads to the activity diagram of Figure 9.3, which is where the “Mutation” object of Figure 9.2 comes to light. There are two starting points to show the independence of the users’ input and the processing of events. As soon as the event hub notifies subscribed systems of the new event in the queue, the controller transforms the event into a mutation of the database. A mutation is a change of state in the database, for instance, an update of the location of a unit. These mutations need to be stored chronologically, and this chronology needs to be fixed. As soon as the user specifies a timeframe to view data from, the mutations will be played back in this fixed order. One could imagine the large volume of data being processed, and if the user would request a moment in far history, many mutations need to be played back. Therefore, the system will include a snapshot system, in which each given time (which depends on the data flux) the current state of the database will be stored. The mutations will be applied to this snapshot to catch up with the specified point in time in order to decrease processing time.

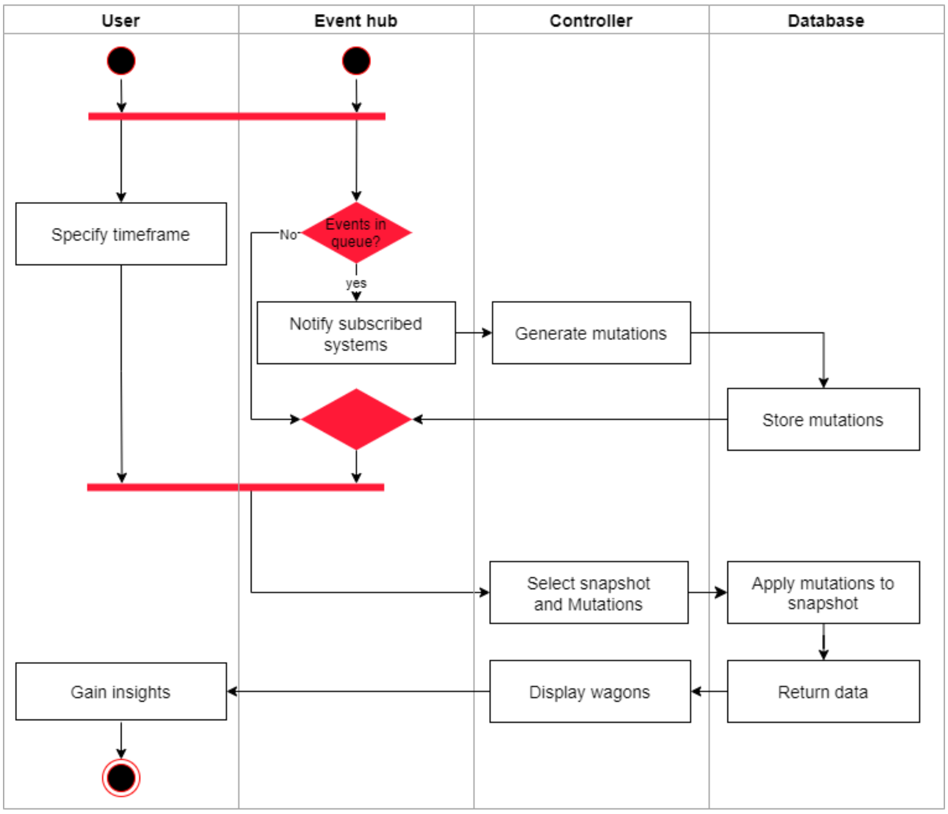


Figure 9.3: Validated event-driven architecture

Only the View aspect of the MVC has been found undesignable at this point in time. It was concluded that there is not a single view that satisfies all stakeholders: each stakeholder has its demands, that despite showing overlap, were too divergent to incorporate at once in a single view.

9.1.3 Reflection on generalisability of the artefacts proposed in this thesis

The central question asked at the end of a design science exercise is about the generalizability of the newly proposed artefacts. Below is the author’s reflection in regard to this. The full architecture that has been proposed is applicable to any rail freight chain that has the four involved stakeholders. This is the context in which the architecture has been designed and validated by observations of case studies that lie in this specific context. Following Wieringa (2014), in the context of the present research, one can not argue for claiming generalizability based on statistical reasoning. Instead, we will follow Wieringa's reasoning based on the similarity of contexts. In line with this, we think that the architecture might well be useful in contexts in which the four stakeholders are not only present but also have similar goals, similar interests, requirements and organizational culture and process-oriented thinking. As the presence of similar contextual factors might well create similar organizational mechanisms across similar but different organizations, it might well be possible to expect that a solution that was suitable for one organization might be possible to be considered suitable also in those different but similar organizations in the same business sector.

9.2 Implications

9.2.1 Implications for researchers

This section discusses how this thesis has a potential impact on research. We translate these implications into specific actions for future research in Section 9.3.

This thesis has crafted the principles of operational digital twins in rail freight that could be built upon, which is an extension of the mostly asset focussed current scientific knowledge. The result of the research is a set of requirements that can be worked out further in order to gradually build on the innovation. The requirements could lay out the foundation for other researchers interested in designing solutions to the rail sector. The requirements in this sense could be considered a theory (Wieringa, 2014) which could be refined based on more empirical research in the future.

The thesis has provided a new perspective of digital twins in this field, where operational processes are the focus. A very important aspect of this new perspective is the way how a twin is viewed: it is more than a graphical interface. But rather, the focus for designing a digital application should be on the underlying back-end: a digital twin is about centralizing information.

Furthermore, the research has given more detailed insights into the processes and the applications in the IT landscape that are related to rail freight. This could be used as related work for any upcoming research projects in the field of rail freight.

Additionally, this thesis adds to the knowledge of Information Systems Research by analysing one particular context which so far evaded the deep exploration on researchers' side. The thesis identified and presented recurring issues in the field of rail freight. The sixfold challenges as presented in Chapter 6 inspire the research field to build on. Using the information in the chapter, we call for more research on the issues we determined to support the development of new solutions in the field.

Finally, the thesis adds to current knowledge on how digital twins could be a solution to *transparency* related issues in operations, rather than on the asset level. Much of the current research is closely related to maintenance and asset management, but this work shows that benefits could be gained in the operations as well. This might be a new perspective for upcoming research with regards to rail freight, or maybe other organisational chains as well.

9.2.2 Implications to practice

How does this thesis affect practitioners in the rail freight chain? The author of this thesis thinks this can be directly related to the core problem statement. The artifact as designed in this thesis is nowhere near complete, but as mentioned before serves as the founding principles of a new innovation. In Section 9.3 will be explained how the artifact should be further developed for reaching its full potential, in this section the potential benefits to practice will be discussed.

First and foremost: the artifact could be a guiding principle of the new proof of concepts as designed by CGI the Netherlands. The Moerdijk and Botlek PoC's show potential, and without a doubt more of these concepts will arise in the near future. If a similar architecture will be used for all upcoming developments, the gradual implementation we propagated in this research will be realised. In this way, a central entity could be created as a solution to transparency related issues.

These transparency related issues are the problems to which this thesis proposes a solution. The gradual implementation of new transparency-related solutions is twofold: it has an impact on the safety on the rails and in the areas near train tracks serving the transport of hazardous materials, and the very needed capacity increase could be realized for modality shifts. The former could be achieved by having new and more accurate knowledge of the whereabouts of hazardous cargo, and the latter could be achieved by phasing out the need for NCBG areas.

Furthermore, an insight that this thesis delivers, is the way the organisations should realise their ambitious innovation. We note that 'Modular' is a recurring term, which applies to many different fronts of the architecture: the number of data sources, the number of sensors, the number of databases, the different types of views, and so on. The more governing aspects will be given in the next section.

9.2.3 Recommendations to practice

We think of the Network Operator as the initiator of the innovation, because this party benefits the most of the system and has the most resources to realize the development, as shown in Section 7.2. Furthermore, the Network Operator is the party responsible for safety on rails and for increasing the capacity. For this system to reach its potential, the Network Operator or a third party such as CGI should be the one to gradually implement such a system. By maintaining the provided architecture as a guideline for upcoming projects, such as Moerdijk and Botlek, the many different proof of concepts could be gradually merged into one system. By using a single architecture for different solutions, the Network Operator could reduce the silo's in which different applications operate and it would make scaling up from proof of concept to working system easier. Recommendations on how to increase the chances of a successful implementation for the Network Operator are given in the "future research" section (Section 9.3).

In the steps of gradually implementing, we recommend that first the proof of concepts in the Netherlands will be designed and integrated as one into the architecture that is proposed in this thesis. It could start very low level: from 1 sensor to a multitude of data sources. Then second, once a solid foundation has been crafted the end-users could be gathered for eliciting design-level requirements. This would be the key to generating the value of the gathered data. And third, as soon as the twin shows results and measurable effects can be observed, the EU could start thinking about an EU-wide solution that stimulates the transfer of goods from the roads to the rails.

Our next recommendation concerns the Emergency Services team. Even though they are the main stakeholder as far as benefits go, they will be less impacted by the innovation as far as development goes. This party is more concerned with how the data is presented and utilized, and should only be involved with respect to defining design-level requirements.

Furthermore, Table 7.1 presented possible solutions of such as system to four more stakeholders. Even though these were discarded with regards to project scoping, the resulting system could have implications for them. For instance, maybe a negotiation between the Network Operator and the Carriers would be an advantage to chain partnerships: Carriers could rethink sharing their data in return for spending less time on registering shunting movements.

9.3 Future work

Future work should include research priorities from multiple perspectives, aimed at both improving the design and extending the scientific context. In this section, the different lines of follow-up research are provided in order of perceived priority. Therefore, it could be regarded as a roadmap. The first priority would be to generalize the research into a broader scope, without altering much on the designs. This design is made in a specific context, but with some extra validation, the design is hypothesized to be applicable in a broad scope. Then a first version of the artifact should be implemented, as low-key as possible. After this, the design could be perfected by eliminating any limitations of this research. Finally, the designs could be extended by designing specific solutions for the stakeholders.

9.3.1 Putting the artifact in a different context

The design that is proposed is for a narrow scope and has a specific purpose. The next research step would be to make the design more generalizable to a broader context, such as extending the scope to rail freight beyond the Netherlands or even the EU. This is considered the follow-up research with the highest priority, as the effort that would be put into researching the artifact in different contexts is low compared to the scientific impact that generalizing the model could have. The solution could be tested for a broader scope on three grounds. Firstly, the four stakeholders that have been incorporated were chosen with the aim to design the View aspect of the system; a requirement that was dropped (requirement 4). However, their stakeholder goals remained the foundation of the design and validation, and hence the solution should be applied to different contexts first, in order to ensure its validity in the broader contexts. Secondly, the high-level architecture was designed for this research context, but might also be considered a scientific extension of the Redelinghuys (2020) framework. With some alterations such as the removal of field-specific nodes, perhaps the model could be generalised to all digital twin architectures. And thirdly, the prior reasoning might also apply to the event-driven architecture of the controller. The solution is field-specific, but with some alterations, the model could be validated to be applicable to any digital twin.

9.3.2 Testing the artifact for the first time

This is about realizing the model for the first time. As modularity has been emphasized so much in this project, the system could be realized by first implementing the architecture with a single sensor, a single RESTful source, and a single message-mechanism source (third-party source connected to the event-hub). From these few individual sources, a proof-of-concept could be built using a simple relational database (requirement 2) and by implementing a controller that processes the events as dispatched by the event hub. Because much time was spent on the assumption that a GUI would be a great contribution by this thesis, no time was left to implement a first version of the architecture. This would be a very good next step in science, as an implemented version of the model provides new ways of validation and expansion. The effectiveness, modularity and scalability of the system could be assessed when an actual version is implemented

9.3.3 Addressing the limitations of the artifact

In future research, more limitations could and should be eliminated by performing additional follow-up studies. The previous sections did address future work that helps in eliminating some concrete limitations, but not all of them. Applying different contexts (broadening the scope) helps eliminate the bias that might come from the fact that only the Dutch railway has been studied. Variations between different freight chains may lie in the technological developments of a country. You could imagine a more developed country having a more supportive system at its disposal than a country focussing more on keeping costs as low as possible. In different contexts, the IT landscapes may look different than the ones provided in this research, even though some use of abstraction was made. Implementing the system for the first time will help eliminate the limitations as a result of the technological assumptions or the used validation methodologies. But there is more to find out: how could independent wagon tracking and identification be best realized? Or is the event-hub indeed capable of handling both messaging mechanism sources and a wide range of sensors?

9.3.4 Building on the artifact

This could be done by engineering the design-level requirements or by adding new interesting innovations to the technology. By eliciting the design-level requirements, specific solutions could be designed on the basic foundation as a result of implementing the first version of the artifact. This would imply that the stakeholder goals could be assessed in detail: what do they need to know? How could this be translated into a view? What sensors do these solutions need? This is where this research went beyond its own scope, but the design that is proposed is compatible with independent solution designs.

9.4 Limitations

This is the last section of this thesis, in which we discuss the limitations of the research as a whole. We acknowledge that throughout this project, there were made some assumptions that might affect the presented models and designs in a different artifact, even though the scope of the population has been carefully expressed.

First of all, all models and designs are based on the case study of the Dutch rail freight sector. The assumption made here is that, mainly due to its cross-border nature, chains in different countries are similar. For instance, the operational sequence diagram (Figure 4.1) and the IT landscape (Figure 5.1) have been heavily based on interviews in the case study. There have been attempts to map most of the findings to scientific literature in order to make the models generalisable. However, some detailed processes, communication flows and IT systems could not be completely backed up by literature. This implies that the models might have some deviations in other contexts and that the designed system may need some alterations in case of small incompatibilities.

Then, some technological assumptions have been made. The model assumes that technology has advanced far enough for automated wagon identification to be stable. Even though the model allows for different hardware setups, automated wagon identification has been incorporated as an important requirement. Also, layer 3 has been assumed to exist as an off-the-shelf solution, in which the sensors of all types could be generalised by a single adapter that is compatible with the event hub. Furthermore, processing power has been assumed to be plenty, and therefore no scalability solutions were officially proposed.

And finally, there are some limitations in the validation stage of this research. For instance, the thesis had to be validated on independent opinions, rather than based on factual data. Even though effort has been made by randomly selecting subjects, by asking unguiding questions and by asking all of the subjects their perceived knowledge about a certain subject, their answers still might have some bias due to their roles in the Dutch freight chain.

Resources

Aldea, Iacob, Hillegersberg, v., Quartel, Bodenstaff, & Franken. (2015). Modelling strategy with ArchiMate. *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, 1211-1218.

Almotairi, B. F. (2011). Information flows supporting hinterland transportation by rail: Applications in Sweden. *Research in Transportation Economics*, 15–24. doi:https://doi.org/10.1016/j.retrec.2011.08.003

Anwar, M. (2019). Digitalization in Container Terminal Logistics: A Literature Review. *27th Annual Conference of International Association of Maritime Economists*, 1-25.

Anylogic. (2020). *Alstom Develops a Rail Network Digital Twin for Railway Yard Design and Predictive Fleet Maintenance*. Retrieved from anylogi.com: https://www.anylogic.com/digital-twin-of-rail-network-for-train-fleet-maintenance-decision-support/

Bontekoning, Y. M. (2004). Is a new applied transportation research field emerging?—A review of intermodal rail–truck freight transport literature. *Transportation Research Part A: Policy and Practice*, 1–34. doi:https://doi.org/10.1016/j.tra.2003.06.001

Cambridge Systematics, Inc. (2007). National Rail Freight Infrastructure Capacity and Investment Study.

Čerka, P. (2015). Liability for damages caused by artificial intelligence. *Computer Law & Security Review*, 376-389. doi:https://doi.org/10.1016/j.clsr.2015.03.008

Cohn, M. (2004). *User stories applied: for agile development .* Addison-Wesley.

Den Hengst-Bruggeling, M. (1999). Interorganizational coordination in container transport: A chian management design.

Dick, J., & Hull, E. (2017). *Requirements Engineering.* Springer International Publishing.

Dijksma, S. (2016, 5 26). Vervoer gevaarlijke stoffen per spoor. Den Haag.

Directorate-General for Mobility and Transport. (2019). *Transport in the European Union.*

Ebeling, C. W. (2009). Evolution of a Box. *Invention and Technology 23, no. 4*, 8–9.

Eurostat. (2018). *Modal split of inland freight transport*. Retrieved from europa.eu: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Freight_transport_statistics&oldid=496663

Eyre, J., & Freeman, C. (2018). Immersive Applications of Industrial Digital Twins. *EuroVR 2018.*

F. Tao, H. Z. (2019). Digital Twin in Industry: State-of-the-Ar. *t. IEEE Transactions on Industrial Informatics*, 2405–2415. doi:https://doi.org/10.1109/tii.2018.2873186

FEDeRATED. (2021). De DTS en de Basis Data Infrastructuur voor. *Digital Transport & Logistics Forum*.

Fernández L., J. E. (2004). A strategic model of freight operations for rail transportation systems. *Transportation Planning and Technology*, 231–260. doi:https://doi.org/10.1080/0308106042000228743

Fowler, M. (2003). *UML Distilled 3rd edition.* Addison-Weasley.

G. Eliza, A. N. (2013). ICT for Cooperative Supply Chain Visibility within a Port Centric Intermodal Setting: The Case of the Thessaloniki Port-Rail-Dryport Integration. *International Journal of Advanced Logistic*.

G. Marchet, A. P. (2009). An exploratory study of ICT adoption in the Italian freight transportation industry. *International Journal of Physical Distribution & Logistics Management*, 785–812. doi: https://doi.org/10.1108/09600030911

Gago, S. (2018). ICT in the Polish Railway Industry. *Problemy Kolejnictwa - Railway Reports.*, 75-81. doi:10.36137/1792e.

Geiger, C. (2016). ICT in Green Freight Logistics. *Green Transportation Logistics*, 205–241. doi:https://doi.org/10.1007/978-3-319-17175-3_6

Glickman, T., Erkut, E., & Zschockec, M. (2007). The cost and risk impacts of rerouting railroad shipments of hazardous materials. *Accident Analysis & Prevention* (pp. 1015-1025). Elsevier.

Grieves, M. (2014). Digital Twin: Manufacturing Excellence through Virtual Factory Replication.

Grieves, M. (2016). Origins of the Digital Twin Concept. doi:10.13140/RG.2.2.26367.61609.

Heller, C. (1999). Tracking & tracing in combined road/rail freight transport. *IPTS Technical Report Series*.

IEEE. (1998). Standard for Application and Management of the Systems Engineering Process. *IEEE Std 80 1220-1998*.

Islam, Z. (2013). The potential of alternative rail freight transport corridors between Central Europe and China. *NewRail Centre for Railway Research*, 45-57.

James Rumbaugh, I. J. (1999). The Unified Modeling Language for Object-Oriented Development. *Reference manual*.

Josey, A. (2016). *TOGAF® Version 9.1-A Pocket Guide.* Van Haren.

Josey, Lankhorst, Band, Jonkers, & Quartel. (2016). *An Introduction to the ArchiMate 3 specification.* The Open Group.

Ke, G. (2020). Managing rail-truck intermodal transportation for hazardous materials with random yard disruptions. *Annals of Operations Research .*

Kennisinstituut voor Mobiliteitsbeleid. (2019). Benuting Multimodale Achterlandknooppunten.

Kia, M., Shayan, E., & Ghotb, F. (2020). The importance of information technology in port terminal operations. *International Journal of Physical Distribution & Logistics Management*.

Kitchenham, B. (2004). Evidence-Based Software Engineering. *Proceedings of ICSE 2004*, 273-281.

Knopf, J. W. (2006). Doing a Literature Review. *Political Science and Politics*, 127-132.

Kohn, W. R. (1980). *Washington, DC: U.S Patent No. 4,230,061.*

Krasner, G. (1988). A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System. *ParcPlace Systems, Inc.*

Kritzinger, W. (2018). Digital Twin in manufacturing: A categorical literature review and classification. *International Federation of Automatic Control*, 1016-1022.

Lauesen, S. (2002). Styles and techniques. In *Software requirements.* Addison-Wesley.

Lee, J. (2015). A cyber-physical systems architecture for Industry 4.0-based manufacturing systems. *Manufacturing Letters*, 18-23.

Liu, X., Saat, M. R., & Barkan, C. (2013). Safety Effectiveness of Integrated Risk Reduction Strategies for Rail Transport of Hazardous Materials. *Journal of the Transportation Research Board.* Washington: Transportation Research Record.

Lucassen, G., Dalpiaz, F., van der Werf, M., & Brinkkermper, S. (2016). The Use and Effectiveness of User Stories in Practice. *22nd International Working Conference, REFSQ 2016*, 205-222.

Mann, S. (2016). The research interview. *Reflective practice and reflexivity in research processes*.

Marinov, M. (2018). Sustainable Rail Transport. *Proceedings of RailNewcastle Talks 2016.* Newcastle: Springer.

Matthijs Otten, E. T. (2020). Outlook hinterland and continental freight 2020. *Topsector Logistiek*.

Meng, S.-M. (2010). Criteria for services of air cargo logistics providers: How do they relate to client satisfaction? *Journal of Air Transport Management*, 284-286.

Ministerie van Infrastructuur en Waterstaat. (2018). *Maatregelenpakket Spoorgoederenvervoer.*

Ministerie van Infrastructuur en Waterstaat. (2019). Goederenvervoeragenda.

Ministerie van Verkeer en Waterstaat. (2004). Besluit van 3 december 2004, houdende vaststelling van voorschriften met betrekking tot de bekwaamheid en geschiktheid van spoorwegpersoneel. *Staatsblad van het Koninkrijk der Nederlanden.*

Moogk, D. R. (2012). Minimum Viable Product and the Importance of Experimentation in Technology Startups. *Technology Innovation Management Review*, 23-26.

Moonen, H. (2018). Gevaarlijke stoffen juist op het spoor. Rotterdam.

Moonen, H. (2021). Digital Twins zijn cruciaal bouwblok voor toekomst railgoederenvervoer. *Chemische Logistiek Magazine.*

Mutlu, C. E. (2015). *Making Things International 1: Circuits and Motion* (Vol. Containers). University of Minnesota Press.

Nunix. (2020). *How Digital Twin Technology is Helping Build a Smart Railway System in Italy*. Retrieved from nutanix.com: https://www.nutanix.com/theforecastbynutanix/industry/how-digital-twin-technology-is-helping-build-a-smart-railway-system-in-italy

Perego, A., Perotti, S., & Mangiaracina, R. (2011). ICT for logistics and freight transportation: a literature review and research agenda. *International Journal of Physical Distribution & Logistics Management*, 457–483. doi:https://doi.org/10.1108/0960

Port of Moerdijk and ProRail. (2020). *De journey van goederentreinen van en naar de haven van Moerdijk.*

Port of Rotterdam. (2019). *Feiten en Cijfers.*

ProRail. (2018). Spoorgoederenvervoer op Kijfhoek.

Provan, K., & Kenis, P. (2007). Modes of Network Governance: Structure, Management and Effectiveness. *ournal of Public Administration Research and Theory*, 229-252.

Redelinghuys, r. (2020). A six-layer architecture for the digital twin: a manufacturing case study implementation. *Journal of Intelligent Manufacturing*.

Responsible Care, ECTA, Cefic. (2011). Guidelines for Measuring and Managing CO2 Emission from Freight Transport Operations.

Rijksoverheid. (2019). *Vervoer van gevaarlijke stoffen via het spoor* . Retrieved from Rijksoverheid.nl: https://www.rijksoverheid.nl/onderwerpen/goederenvervoer/vervoer-van-gevaarlijke-stoffen-via-het-spoor

Roso, V., Woxenius, J., & Lumsden, K. (2009). The dry port concept: connecting container seaports with the hinterland. *Journal of Transport Geography*, 338–345. doi:https://doi.org/10.1016/j.jtrangeo.2008.10.008

Rosova, A., Balog, M., & Šimeková, Z. (2013). The use of the RFID in rail freight transport in the world as one of the new technologies of identification and communication. *Acta Montanistica Slovaca*.

Schott, D. L. (2018). Break Bulk and Bulk Terminals. *Encyclopedia of Maritime and Offshore Engineering*, 1–10. doi:https://doi.org/10.1002/9781118476406.emoe195

Schroeder, G. (2016). Visualising the digital twin using web services and augmented reality. *IEEE 14th International Conference on Industrial Informatics*, 522-527.

Siemens. (2020). *Siemens Mobility levert Digital Twin Testing System voor HSL-Zuid*. Retrieved from mobilitymatters.siemens.nl: https://mobilitymatters.siemens.nl/rail/hsl-zuid/

Sonnenberg, C., & Sehested, H. (2011). *Lean Innovation: A Fast Path from Knowledge to Value.* Springer-Verlag Berlin Heidelberg.

Stargel, E. G. (2012). The digital twin paradigm for future NASA and U.S. air force vehicles. doi:10.2514/6.2012-1818

Surianarayanan, P. R. (2020). Digital twin: The industry use cases. In Advances in Computers. *Advances in Computers*, 285–320. doi:https://doi.org/10.1016/bs.adcom.2019.09.006

TIBCO. (2017). *Dutch Railways Delivers More On-time Train Services with TIBCO*. Retrieved from tibco.com: https://www.tibco.com/customers/dutch-railways

TNO. (2018). *Leidt mobiliteitstransitie tot reversed modal shift voor goederenvervoer?* Retrieved from TNO.nl: https://www.tno.nl/nl/aandachtsgebieden/mobiliteit-logistiek/roadmaps/smart-and-safe-traffic-and-transport/societal-impact-for-accessibility-and-liveability/mobiliteitstransitie/mobiliteitstransitie-omgekeerde-verschuiving-goederenvervoer/

UIC. (2008). Unaccompanied Combined Transport Guide on Coding and Certification.

Ulianov, C., Hyde, P., & Shaltout, R. (2016). Railway Applications for Monitoring and Tracking Systems. *Proceedings of RailNewcastle Talks*.

van Gompel, M. (2016). *Registratie gevaarlijke stoffen op emplacementen nog steeds ondermaats*. Retrieved from SpoorPro: https://www.spoorpro.nl/goederenvervoer/2016/05/31/registratie-gevaarlijke-stoffen-op-emplacementen-nog-ondermaats/?gdpr=accept&gdpr=accept

Vannieuwenhuyse, B., & Brouwers, E. (2020). Ontwikkelen van governance model voor 4C initiatieven. *Topsector Logistiek*.

Verwater, E. (2020). *Rangeerterrein haalt blustest niet; wagons met gevaarlijke stoffen nog steeds niet toegestaan*. Retrieved from ad.nl: https://www.ad.nl/rotterdam/rangeerterrein-haalt-blustest-niet-wagons-met-gevaarlijke-stoffen-nog-steeds-niet-toegestaan~ac8cd0e1/

Vis, I. F. (2003). Transshipment of containers at a container terminal: An overview. *European Journal of Operational Research*, 1–16. doi:https://doi.org/10.1016/s0377-2217(02)00293-x

Vis, I. F., & de Koster, R. (2003). Transshipment of containers at a container terminal: An overview. *European Journal of Operational Research*, 1–16. doi:https://doi.org/10.1016/s0377-2217(02)00293-x

Watson, & Webster. (2002). Analyzing the past to prepare for the future: Writing a literature review. *MIS Quarterly*, 13-23.

Wiegmans, B., Hekkert, M., & Langstraat, M. (2007). Can Innovations in Rail Freight Transhipment Be Successful? *Transport Reviews*.

Wieringa, R. (2014). *Design science methodology for information systems and software engineering. .* Springer.

Win, A. (2008). The value a 4PL provider can contribute to an organisation. *International Journal of Physical Distribution & Logistics Management*, 674–684. doi:https://doi.org/10.1108/09600030810925962

Wolbers, J. (2013). The Common Operational Picture as Collective Sensemaking. *Journal of Contingencies and Crisis Management*, 186-199.

World Economic Forum. (2009). Supply Chain Decarbonization. *Logistics and Transport Partnership Programme*.

Xie, & Levinson. (2009). Modeling the growth of transportation networks: a comprehensive review. *Netw Spat Econ*, 291–307. doi:10.1007/s11067-007-9037-4

Zacharioudakis, D. L. (2012). *The Blackwell Companion to Maritime Economics* (Vol. Liquid Bulk Shipping). Wiley-Blackwell: Blackwell Publishing Ltd. doi:https://doi.org/10.1002/9781444345667.ch11

Appendix

A. Table of software

System (producer)	Description	Type
Controleapp (ProRail)	System to verify whether manually filled in data on emplacements is correct	
PCS (Portbase)	a large system offering 44 services with different pricing options to at least 12 types of actors/users. The PCS is primarily focussed on information exchange between all the actors in the logistics chain, but extends to almost all types of uses. Services offered to rail operators / haulers by the PCS are among others: customs information sharing, polluting substance declarations, track and trace of cargo within the ports, inspection report insights, loading and unloading reports, payment systems and multimodal check-in systems.	
Navigate (Port of Rotterdam)	This is a booking system for cargo, kind of like booking a passenger trip. On the site it is possible to book voyages from terminal to terminal.	Booking system
Routescanner	Another booking software which is very similar to Navigate.	Booking system
TEUBooker	Booking system for all Dutch hinterland connections through rail and barge.	Booking system
AMRA (Bosch)	Piece of equipment that can easily be applied to wagons and cargo for monitoring cargo. Sensors measure temperature and shocks, provide real time gps data and senses opened doors. The device has two types of connectivities: bluetooth and GPRS.	Cargo information system
BoxInsider (Port of Rotterdam)	This is a container tracker based on event driven messages which are cross checked from multiple sources	Cargo information system
CESAR (CIS)	A search engine for Clients, Forwarders, rail Carriers, Terminal Operators to track and trace loading units. It is a searchable database giving statuses of cargo (ready to pick up, loaded, etc.), not a displayed real-time map.	Cargo information system
Smartbox (DBSchenker)	Software to enable IoT connected services for tracking of cargo conditions. It also protects cargo by installing alarms and trespasser detection sensors.	Cargo information system
ORFEUS (Raildata)	Management system for consignment notes. This information is made available to all parties involved in the shipment of that specific cargo and wagon.	Consignment note system
RMS/FENIKS (ProRail)	System for reservation and management of traffic in NCBG areas, in which regular train tables are not applicable.	Emplacement reservation system
ECOPMS	A system that aims to innovate in the intermodal transport branche, in which the different modalities are transparent and traceable. The platform aims to be a control tower for the flow of information between the different modalities. The platform is aimed on manufacturers and Forwarders, for being able to create the most cost optimal and environment friendly	Information management system

	transport options. This solution aims to optimize individual performance.	
GMS (ProRail)	Portal that offers services provided by other internal software to third parties.	Information management system
Cargostream	This is a system more focussed on Client-Carrier relationships. It provides a single channel communication system for transport orders, inventory management, warehouse flows and logistics transparency.	Order to cash system
RCS (Ab Ovo)		Order to cash system
ORMAS (ProRail)	Order portal for booking routes. Booked routes are forwarded to PRL and are inserted accordingly.	Path coordination system
PCS (RNE)	aims to optimize international rail path coordination. Provides timetabling and automated communication for path requests. While planning is a part of the features of the system, its general goal is to harmonize path requests along the international chain and make the operations transparent	path coordination system
TMS (Thales)	provides control to the rail network by automatically setting routes and solving conflicts in planning as it tracks train movements.	Path coordination system
TOS (Cofano)	An order-to-cash like system for terminals. These applications support and optimize terminal operations, from order bookings to the entire physical handling of the cargo and invoicing.	Terminal operating system
Axlecounter (Thales)	System that automatically counts the number of axles passed. If the first sensor counts as many axles as the second sensor, then no train is on the section. If the second sensor has not counted as many axles, then the train is still (partially) on that specific transaction.	Train detection device
Short circuit (GRS)	A sensor that senses the presence of trains through energy pulses.	Train detection device
PRL (ProRail)	System for dispatching trains. Whenever a train is departing its route has to be inserted. Inserting the route is done through PRL.	Train dispatching system
OnTrack (Port of Rotterdam)	Solves the problem of many rail Carriers having their own planning data in silos by providing a single point of access. Users of the software can track trains by having access to planning data of Carriers and event driven data such as loading and unloading data. The software optimizes planning operations, but its main goal is to orchestrate data exchange between logistics parties.	Train information system
TIS (RNE)	System that tracks trains on the rail network in Europe and shows timetables, delays, forecasts. This is a clear example of fleet and freight management. PCS and TIS are both owned by RNE, therefore data of both platforms can easily be accessed, enabling data insights about delays, dwell times, punctuality development, etc.	Train information system

Toon (ProRail)	System for viewing historical train movements.	Train information system
TROTS (ProRail)	System for tracking and observing trains in real time.	Train information system
UIS (ProRail)	System that calculates delays of flowing traffic.	Train information system
VIEW (ProRail)	A more accurate train tracker for checking track section, trainnumbers and plannings. The application provides visual information to traffic control about track changes and train locations.	Train information system
ASTRIS (ProRail)	System that checks whether a inserted route is safe, and whether track changes can be switched.	Train protection system
ERTMS	this system aims to unionize the more then 20 different train control and command systems across Europe. The system calls itself a Train Protection System, which uses remote sensors to guide safe operations and reduce human errors. ERTMS acts as a kind of standardization of signalling for railway management and interoperation. This could be classified as supply chain execution. ERTMS: In nederland heb je grotendeels ATB beveiliding. Dus die bepaald voor een trein of die ergens mag rijden. ASTRIS gebruikt alle inputs van wissels en seinen om te bepalen of een sein mmag worden doorgegeven. Mag dit op groen, of rood? ERTMS is een techniek in de plaats voor ATB, dus meer gericht op virtuele seinen.	Train protection system
Donna (ProRail)	Timetabling system for train traffic for the longer term.	Train timetabling system
Paloma (Ab Ovo)	System for planning and dispatching trains, personnel and locomotives.	Train timetabling system
VOS (ProRail)	System used for timetabling adjustments for maintaining traffic flow in cases of unforeseen events.	Train timetabling system
Ident (Hastema)	This is a system for automated wagon scanning. Through RFID, or AI driven UIC reading cameras the system scans wagons driving past the scanner. Other features are reading of danger labels, status documentation, alarms for open doors or faulty bottom, wagon types etc. This information could be used for integrating automation for existing pieces of software.	Wagon detection device
DRA (ProRail)	Shunting assistant for the Train Operators for registering shunting movements in NCBG areas.	Wagon information system
ISR (Raildata)	tracking and tracing system for rail traffic across Europe. This goes one step further then TIS in the way that this goes into the detailed scope of tracking wagons, wagon status and wagon performance. This system however lacks the real-time view of TIS in the	Wagon information system

	sense that this system is event-driven and message based.	
Wagon intelligence (DBCargo)	System that tracks all wagons	Wagon information system
WDI (Raildata)	A low budget solution for small and medium Carriers to help them fill in railway undertakings. The focus in the application is primarily on Hermes 30 messages regarding train composition. These messages are shared across the parties in the chain needing the information such as partner Carriers.	Wagon information system
WLIS (ProRail)	Information system for showing real time data about wagons with an emphasis on polluting substances. In CBG areas tracking is mostly automated, but in NCBG areas the data is generated manually.	Wagon information system
Zedas cargo (Zedas)	A very extensive rail cargo system with three main capabilities: shunting support, mainline traffic support and port traffic management. The overall goal of the system is to track incoming wagons, to support planning and to automate invoicing. The system supports integration of IoT services such as automatic wagon detection and weighing systems. The system does not seem to support actual tracking and tracing of wagons, but is meant to register the trains arriving and leaving.	Wagon information system

B. Interview Consultant I

Occupancy	IT consultant CGI
Date	03-03-2021
Duration	1 hour and 15 minutes
Setting	Microsoft Teams

Introduction

The subject has had many different jobs, most of them in the logistics field. Has experience in software development for different purposes, such as robotics systems at terminals. Currently architect and data analyst. Also many project in the rail sector, software with regards to trains. Also many projects in rail freight, working with and for the Network Operator in the Netherlands. The subject shows model of the rail infrastructure, tracks, areas, signals, etc.

Important software in rail freight

PRL and VOS are important. As soon as there is a planned route you have both steering and timetabling adjustments: you have a plan and whenever you have discrepancies of the planning you have to correct. This could be done by either having a train speed up, or by delaying another train. This is steering, but whenever a train has to be cancelled or you have to get new personnel in action, that needs timetabling adjustments. These timetabling adjustments are in VOS and minor corrections are in PRL. PRL is for the traffic control, who inserts routes for trains. Inserting routes means: adjust track switches accordingly to reserve the path for a certain train. Checking whether insertion is safe is done through ASTRIS. ASTRIS uses interlocking systems to make sure track switches are safe to be changes. Traffic control plans from service control point to service control point, which is an area between the free tracks. Such a point could be a station, a junction, a bridge, emplacements and so on.

For planning we have the system DONNA. In the ProRail part of DONNA timetables are generated. These timetables are loaded into VOS for the timetabling adjustments. This is daily. DONNA has three parts, material, personnel and services. The first two are only used in passenger transport. In logistics the Carriers have their own systems for personnel and material.

Then we have UIS, which is used for calculating delays. So for instance if you are at a platform and on the digital board it is given that a train is delayed, that comes from UIS. But it is also relevant for freight trains.

Then for train compositions and dangerous goods there is WLIS. This system is used for tracking and tracing of wagons containing these materials. The inputs of wlis are the arrivals and departures of wagons. For shunting movements we have DRA, the shunting assistant. Train drivers are obliged to register all shunting movements within 5 minutes through the DRA. Note, these are shunting movements, not train movements. Train movements are automatic, these are in the centrally governed areas, whereas shunting movements are not. So you have WLIS which is for tracking trains, and wlis track occupation which is for registering track compositions. WLIS is on train level, wlis track occupation is on wagon level, but only for dangerous goods. Also non-hazardous goods can be registered in WLIS, but that is not obliged. So in WLIS all wagons containing polluting substances are registered, but it is not given which wagons are in between them. This makes searching for the right wagons quite intricate. In WLIS, Carriers can only view their own wagon contents, they cannot view those of their competition. The data in WLIS is provided by the carries themselves.

There is also TROTS for tracking and tracing of trains in centrally governed areas. A track is composed of track sections. On these sections some axlecounters or short circuit sensors are installed. These sensors detect trains and provide TROTS whether a train is on a certain track section. In TOON this information is stored as historical information.

Third parties making use of WLIS

Well, WLIS is a system without screens, it rather is a backend. For entering data there is GMS, the portal made for parties who do business with ProRail. So a piece of that is WLIS. GMS is meant for information disclosure with third parties with a goal of chain harmonisation. If you want to know more about GMS you have to ask *** (consultant II).

Terminal systems

For terminals the job is easier: trains decide the rhythm. Trains are planned, but the tricks at terminals is about the ships. Cargo at terminals is stacked and stored, often through automated straddle Carriers. These Carriers drive the cargo from the stacks to the rail terminal. Each terminal has its own systems. But there are standardisation methods such as the EDIFACT message. Maybe you can research Portbase, they provide a lot of digital messaging systems within ports.

There is the “controle app” which is sometimes used at terminals to see whether the information provided by Carriers is correct. This app works with WLIS track compositions.

Lack of digitalisation

Regarding freight traffic on emplacements: you just don’t know what happens there. You have to completely rely on the information that Carriers supply you with. If they are lacking effort you are in bad luck. All cargo at emplacements get handed over to different Carriers. This implies also taking over full responsibility of the wagon. This really toughens transparency.

Digital twins in rail freight

Making a DT of rail freight traffic is hard because there are many involved parties, and these parties often do not have the money to invest. Margins in rail freight are quite low. And all the Carriers are each other’s competitors. There is a lot of distrust, and they are not keen of others snooping around in their data. Many Carriers think of the rail operator of an authoritarian party, they think the Network Operator charges too much.

The growth opportunities are in transparent information, but should be protected. Transparency of information is the key issue right now. Right now we have many parties handing over information, and sometimes this is inaccurate. Because of the distrust the twin should have a security system. For instance, that prorail can only access wagons containing polluting substances. And for the Carriers on wagon level, that they can see everything about their own fleet.

The issue is, Carriers mostly want money, and rather not invest it. They are not keen on centralisation. They might be hesitant towards a DT. To be honest, I see it having very low odds of success.

C. Interview Consultant II

Occupancy	IT consultant CGI
Date	03-03-2021
Duration	1 hour and 10 minutes
Setting	Microsoft Teams

Introduction

I think that **** (Consultant I) and I have similar profiles. I have done some tests with sensors at Moerdijk, and I think the work I have done there is quite hands-on. So, that is a job on the technical level, but I am also a product owner for the WLIS chain, and in that project I do not do any programming.

The digital twin in rail freight

So, what we do at Moerdijk is on a very detailed level, on these emplacements it is about tracking and tracing of wagons. Regarding Digital Twins, I think this innovation finds its value in the wider chain. We have systems such as TROTS and ASTRIS that track wagons, and at Moerdijk we have axle counters because these systems don't work at those areas. Here camera's perform identification of wagons, and the axle counter follow the wagons. Now you know where a wagon is, but not why it is there. It is useful to fire departments, but if you want to add value for ProRail, then capacity management is way more important. And for that information, you are not just in the systems of ProRail, but also on those of partners such as Port of Rotterdam or Moerdijk. These have the systems to tell what cargo is transhipped onto what boat, and when incoming cargo is bound to be loaded onto trains. Besides, TROTS is on CBG, centrally serviced areas. But with freight trains the most complex operations happen on NCBS, the non-centrally service areas. This is where it gets interesting for a DT: no TROTS, no PRL, no ASTRIS.

The processes of Carriers are very untransparent. They are transporting the goods from A to B, but B is not always the destination. The destination is transporting the goods multimodally. But you have many intramodal Carrier changes, in which liability is handed over completely. For instance the train from Moerdijk to Kijfhoek is with a different Carrier than the Carrier transporting it from Kijfhoek to Italy. And that is not just one transporter, but could also be seven. And in between trains get rearranged and then you lose track of all cargo. You see, freight trains are different from passenger trains. Regarding passenger trains you transport people from a to b, but in logistics the train is just a part of a larger chain.

I think especially the terminals will gladly see a digital twin. The capacity management features will get a lot of attention within ProRail. With freight trains departure is very unpredictable. If a ship is delayed, say because of wind, the train won't leave without cargo. Maybe it will leave one or two hours later. That makes for capacity issues. Especially at emplacements track sections are reserved through TSR for way longer than needed. And for a larger track section than needed. This is where a DT would have a lot of value. Because if a Carrier reserves TSR1, they actually block all of Moerdijk because that is the entrance. During the time no train can enter or leave.

We do have the virtual train for passenger transport, in which the location of a train is the key. But with respect to a digital twin for freight train, I think it is most important to twin the contents, and not so much the train.

Controle app

The controle app is administrated by ProRail. Train drivers use DRA to register shunting movements, and the controle app is being used to verify whether the information entered is correct. Employees of ProRail physically drive towards the tracks to check if all wagons are positioned as registered in WLIS.

This app will definitely stay relevant. Given any confidence level of the system, there still has to be a check to certify the information correctness. This will get a very political story. Somewhere there will be made a calculation: how much does a human life cost? So they will make a balance between the odds of a wagon not being scanned explodes, and the costs of physically travelling to the emplacements. There also is a danger that when a Carrier gets to know of the sensors, they would love it! But what if something goes wrong, and a wagon has not been registered by the system, who is liable?

Culture problems in the chain

Regarding culture: on board level there are a lot of arguments. But Carriers themselves are quite positive about the innovations of ProRail.

But I do see a challenge regarding DT's. It is not possible yet to get the location of a locomotive. And even if Carriers have GPS fitted on their fleet, why would they share it with ProRail? What is in it for them? There is no central registration for fleet assets. So you have to do it differently. Without the use of cameras you get the WLIS principle. If you do add the cameras you could make a small check whether some WLIS information is incorrect.

And also there you have liability issues. If you tell a Carrier that his information is not correct, he could blame the previous Carriers. He has gotten wrongful information out of Italy.

Train safety

You have ERTMS, which in the Netherlands currently ATB security. That decides whether a train is allowed to drive somewhere. ASTRIS uses all inputs to check all tracks changes and signals to confirm whether a signal may be given. It confirms if a signal may be set to green or red for example. ERTMS will take over ATB and is aimed on making these signal virtual. But both will not be very relevant regarding a Digital Twin. Except that maybe somewhere in the future, locs will be fitted with GPS, but that is not yet.

Furthermore, you have department incident management within ProRail. If a fire has been reported, incident management calls the Emergency Services and provides the information as given in WLIS. It used to be that all wagons had to be reported, but this is changed to jus wagons containing polluting substances. But if on one track is a wagon with hazardous materials, then the entire track has to be registered. So it comes down to basically all wagons being registered, but it sometimes shows that a track is not occupied by a wagon containing hazardous materials, then you know nothing about that track. WLIS helps performing risk assessments in case of emergency, but you cannot see the distance between the wagons.

Verification of an early version of Appendix G

I think UIS is still relevant to rail freight. UIS belongs to VOS. VOS has a few component, of which UIS is one. Other components are PLAN, User interface and something called messages. That is where the check-in and the order processer are stored. And that on its turn has a link with the order portal. TROTS provides track and trace information to UIS. UIS also has the planning data. So UIS knows about trains how much they are delayed and in which way they travel, but does not know exactly where. TROTS knows about track sections, but not exactly where a train is.

WLIS does make use of TROTS, but this link is very indirect. At CBG WLIS could use TROTS to track and trace trains. On emplacements this is another story, that's when WLIS has to be used manually. In WLIS, Carriers are obliged to report train compositions: what they transport to where, this is including the consignment notes.

In your model it is confusing that WLIS is a service, GMS is a system and Order Portal is a service. WLIS is composed of WLI and SBC, WLI has train compositions and SBC is about track compositions. Maybe you could change Orderportal to ORMAS, the system for ordering or changing routes. You could also adjust timetabling in ORMAS, so that goes into VOS through ORMAS. So ad-hoc changes are done in ORMAS and adjustments in future flows is in DONNA.

RMS/FENIKS is missing in the picture. FENIKS is the new version of RMS, the system that was made for Kijfhoek to see what happens at NCBG. Kind of the same as WLIS, but whatever. WLIS is about what is happening now, and RMS is about what is about to happen. TRS requests are managed using RMS. A TRS is requested through RMS. This is not in GMS, but is a standalone app. So RMS does not give any information to other applications, because it is only at emplacements. RMS is for traffic control and Carrier with the goal to confirm whether a train could enter Moerdijk. This is not possible in WLIS, because in WLIS is only given what is on the tracks now. RMS is about what train is allowed to enter. RMS does have DONNA and VOS timetables, because RMS is the entrance for NCBG. If a train is not allowed to enter Moerdijk because another train is operating there, then we have to get a TSR for the waiting train. This is managed in RMS.

In RMS there is another process called GTI, the check-in system. Freight trains are unpredictable, therefore they first have to check-in before departing. So if a ship is delayed, and the train as well, the train gets into a queue. And when the train wants to depart the train drive has to check in first. If you could find relationships between ships and trains, you could estimate delays. This is where you could create value with digital twins.

D. Interview Terminal Operator

Occupancy	Port operator Moerdijk
Date	23-03-2021
Duration	20 minutes
Setting	Microsoft Teams

Introduction

The subject is short of speech and in need of time, no proper introduction was done but he went straight into the matter.

Communication with chain partners

We have with all line Carriers contractual agreements in which we arrange what information we exchange when. That is defined in very detailed level, which data we need to get from them, but also the method how and on which events that needs to happen. Say a ship leaves the port, then we get an unloading list from the shipbroker, this is often digitally. This is done using EDIFACT, a standardisation method for information sharing which you see in the entire logistics chain.

But for trains we also have these contractual agreements, these are even more detailed because trains are more inflexible. A rail Carrier operates on a trajectory, say Segrate-Moerdijk. He has many customers that reserve containers from Segrate to Moerdijk. As soon has the trains are loaded and ready to depart the send an EDIFACT message, but through XML. This includes: the load, the weights, container numbers, wagon types, contents, and what will happen to the containers. We have nothing to do with consignment notes.

We don't make the transhipment planning ourselves. We do not have any contact with the owners of the cargo. The rail operators do have that contact, they know what cargo has to be loaded onto wat ship. We only execute their orders, there is no planning. An example: goods are sold in Italy with destiny Manchester, then goods are shipped from Segrate to Moerdijk. The operator tells us which containers have the UK as final destiny. From these containers they give the hand-in reference when they arrive at Moerdijk. Using the reference numbers on these hand-in references that the cargo is to be loaded onto ship X. So with regards to stacking, these hand-in references are crucial.

These XML files are sent through This could also be done on the FTP server, that is a bit more usual.

By the way, there is a difference between Carrier and rail operator, but most of the times the Carrier is the operator.

We do not check the cargo, that is the rail operator's responsibility. We do check the container for damage.

We barely have any contact with the Network Operator.

Data inaccuracy

It happens that XMLs contain inaccurate data. Especially for the loading of trains and ships this is important. Wrongful data could have large implications.

Whenever there is a delayed train, this get communicated by phone or mail, at least we hope so. I hope other terminals work with re-requesting timeslots, because we don't! We try to arrange with the Carrier to see if the train could be handled in a different timewindow.

All contact with third parties is done through phone or mail.

Information with respect tot he hand-in references is often lacking.

Terminal systems

At our location we do know exactly what cargo is to be found where. This is monitored in the Terminal system. This is our fundamental system, all relevant data is stored in here: weight of the cargo, cargo type, container number, customs status, references, operators, boxoperator, mode of transport, transshipment planning, position, everything.

Tracking and tracing of containers is performed semi-manually. The terminal system proposes a position for stacking the container. As soon as we get an inbound load, the system generates a stacking configuration. The operatir physically handling the container confirms whether the container is a the proposed position, or if it is placed elsewhere.

Sometimes we make use of Portbase software, bu only for customs. Other software of Portbase is not relevant to our terminal.

Digital twin

A digital twin would at most be of value for tracking and tracing a train. But crosschecks, so checking whether the information we have is right, we won't do that.

E. Interview Rail Carrier

Occupancy	Rail Carrier DBCargo
Date	11-03-2021
Duration	1 hour and 5 minutes
Setting	Microsoft Teams

Introduction

Very enthusiastic talker. Starts talking about many projects he has done for process optimisations within DBCargo. Talks about the structure of the company, how DBCargo Nederlands is part OF DBCargo in Germany.

Processes

Carriers work with contracts, based on the wishes of customsers. Customers place a market order, and this request gets send to service design. There we translate the customers needs into a product with a suitable price. The customers requests wagons for his cargo, the cargo that arrives at certain ports or terminals.

So it all starts with the market order. That gets deposed at sales, which is service design. They look in the entire chain what train service is suitable. This will be the product model, and is mapped to a suitable price.

There is a very big difference that you have to understand. With normal passenger transport you are an occasional traveller. We barely not know this flow in freight traffic, you always have a contract there. No contract, no transport. No Ad hoc transport. This would have been possible of the process was not this expensive. The flows in the chain are completely fixed, no producer ever would say “lets transport these fridged today”. That cargo has to be shipped, that is for sure. However, the modality with which that happens is market forces. It is about money, time, intrinsic values, and so on. For instance, air transport is about low masses, but it is a billion dollar business. Expensive things go through the air.

Sometimes you do have larger volumes, such as on holidays. Then the need for wagons is larger. But the contracts are defined with transport volumes. We work with brandwiths with a Bonus Malus agreement. Contracts define the volume that is shipped each year, because capacity is costly. These contracts are renewed each year. Payment is at the end of each year, and the Bonus Malus agreement makes that Client and the Carrier to fulfil their contracts.

Customers know about our timetabling, and we plan our assets on that. They know DBCargo is 10 o’clock at Shell, and at 11 in Vlissingen. Every contract is mapped to the timetables. In these it is agreed on the wagons and the locations of the cargo. Timetables are defined by ProRail. And now comes the level of detail in the contracts: in this the infra needs are estimated, regarding storage capacity, hill capacity and timetabling. Timetabling is very tight, that’s a given, we are with over 14 Carriers and you often use the mixed net as the Betuwelijn is not always the most efficient, so we have to keep in line with them. And if you are delayed you lose your train service, prorail is very rigorous in that: you didn’t check in, you lose the service. You don not get a green signal because your train number is deleted out of all IT systems. And without an ID, you don’t get your route inserted.

If that happens we need to adjust our timetables. We have transport control for that, they look for alternatives for the delayed train. So, you have lost your train service, but you still have the train number which is valid for the whole day. Using this number you could request a new route, and when the new route is inserted you get your green signal.

You get a timeslot at terminals, kind of like with train services, if you don't make the timeslot you lose it. The terminal wants to use his capacity optimally. First you are going to steer the train, lets say the train goes to Waalhaven. When you have all resources, so you hae the train service, the loc, the train driver, and an ETA, then you call the terminal. You wait for the timeslot request when you know your ETA. When arrived at the terminal you always have to get a green signal first.

Digitalisation

For these contracts we have a digital register.

We also have dynamic wagon statuses in our systems. Together with historical statistics on loading times we could estimate when wagons are available again after transhipping at location X. So if two wagons are unloaded at Rotterdam, than we know this using the dynamical wagon status, and using models we get these wagons to our customer.

Customer care and transport controle make use of systems. We have Paloma (planning loc and machinist) of the producer Ab Ovo. Furthermore we make use of RCS (rail cargo system) by the same Ab Ovo. In Paloma we plan all assets and personnel. Dispatching personnel get their information on tables, we call these old fashionably their service card. The service card is a digital to do list which shows the train service the train driver has to execute. It even states whether the train driver has to walk to a loc, or take the car to a post. Furthermore, everyone makes use of Paloma, whether your traffic control, dispatching train driver, dispatching loc, process management, they all work with Paloma because that is where you service is. All assets are in there.

Paloma is linked with RCS, and RCS contains all order data, but also the train compositions that we send to WLIS. All departing trains get a departure check, this is stated by law. Part of the check is basic: your wagon order. The wagon order is the most important. Then you have all safety checks, right lables, and so on This checkreport is send to process management who administrates it in RCS. Process management verifies whether the checks have been performed, and if so he marks it ready in RCS which triggers a WLIS message. Whenever a new train composition adjustments takes place, the new composition has to be reported in WLIS. Traincompositions are based on the consignment notes, CIM. The CIM is made by the customer or Forwarder.

In WLIS you could transfer wagons to other Carriers. A secondary party could be notified that the cargo could be handed over. The acquirer confirms in WLIS that they have indeed taken over the freight, after which the name of the new Carrier is mapped in WLIS.

Also in Paloma all locs are tracked through registration by the train drivers. Wagons are being tracked using GPS sensors. Those devices have their own batteries. Of all wagons in all of Europe we know their positions. This is done from WI (wagon intelligence), a DBCargo system. This is linked with RCS, but not with Paloma. RCS is a fully integrated order-to-cash system: you have the order system, the contract system, booking system, name it. WI is linked with RCS through ORFEUS.

Culture

DBCargo does not do any business with competition, they want to stay in control. Data sharing is not desirable.

ProRail has nothing to do with our commercial activities.

What we see in the Netherlands is that the competition is fierce between Carriers. There is only little cooperation, we do not make use of one another capacity.

Shortcomings of digitalisation

I do not understand that a terminal would not know a train composition. We send the terminal a message. How else is he going to generate a stacking configuration? The train order is crucial to the terminal. The terminal does not want to have his cranes making unnecessary meters. He wants to pre-stack, i.e. he wants to position his trailer, cranes and containers at the place they need to be. And if the traincomposition is wrong, and all the machinery is misplaced, he would not be happy. We send the terminal the same composition documents as we send ProRail, even the actual weights are reported here. What could happen is that the booked units are not very clear.

Communication is very old-fashioned. We mail the train compositions. We perform a check for all inbound trains, this report goes to process management and they mail the terminal. We always indicate which wagons are where. We indicate the train number and the reversing train number. Portbase has a long way to go, I mean it is 2021 and I talk on behalf of all Carriers that it is absurd that we do not have digital message systems. Also notifying that a train is ready to enter the terminal is through phone.

F. Interview Consultant III

Occupancy	IT consultant virtual train
Date	02-04-2021
Duration	20 minutes
Setting	Microsoft Teams

Introduction

The subject has been working for the application “de virtuele trein” as requested by NS, the largest Dutch passenger rail Carrier. The interview is conducted to get to know the ins and outs of the application to find similarities to digital twins.

The application

One major driver for NS was punctuality, and part of punctuality is having the passengers enter the train as fast as possible, that was why the boarding assistant was made. This is a prototype in which a large LED-bar is on the platform which indicates where the train stops, where the doors are, which class the wagon is, whether there is a wheelchair assistant. This system is fed by the virtual train. Also in trains you have the indicators of upcoming stations, or location dependant advertisements.

Also the “reisplanner” has a function with a trainradar with train information: number of wagons, type of train, characteristics. All this information is fed by the backend which is called the virtual train.

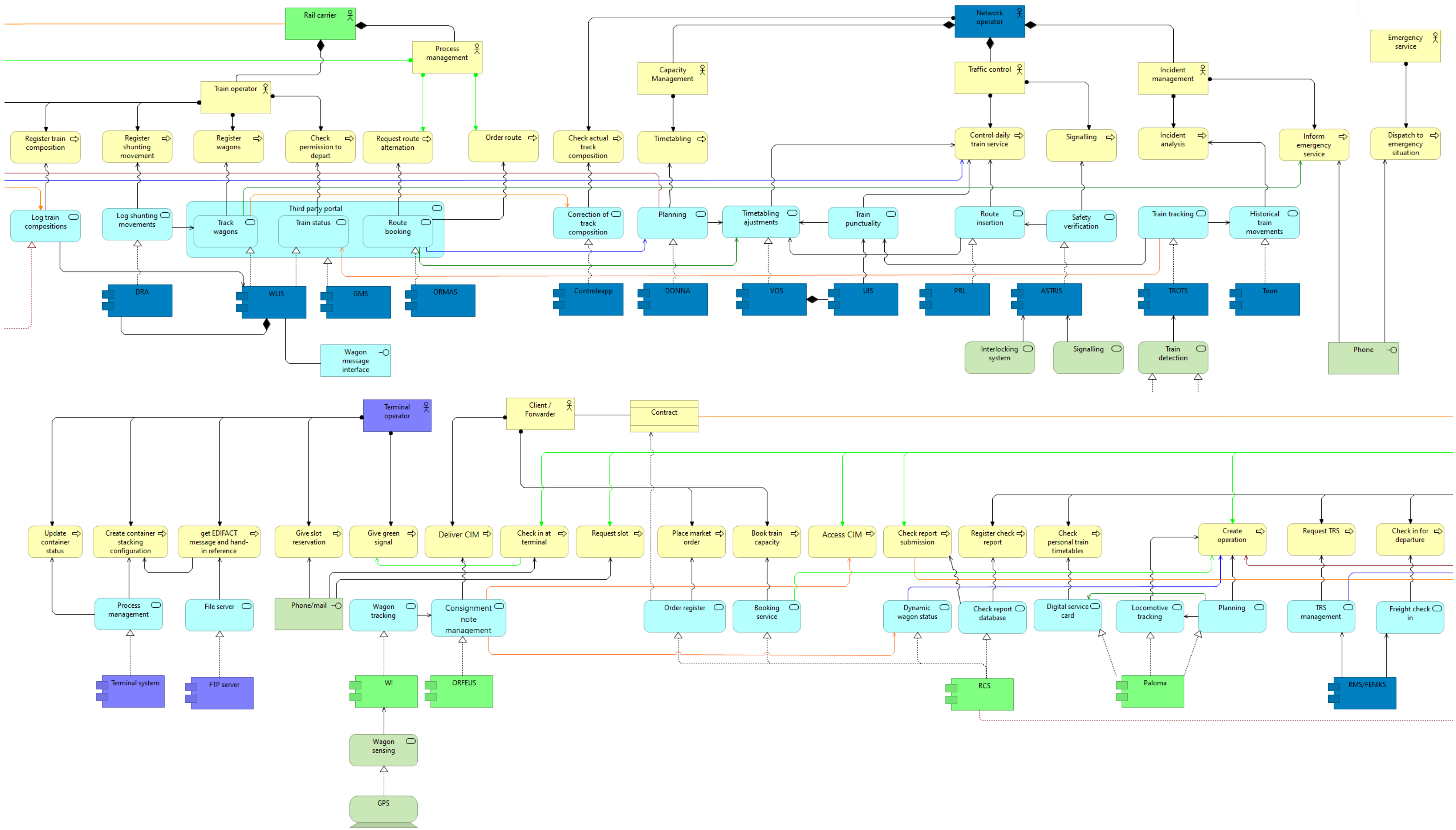
For NS a focus is on unforeseen events, in which data used to deviate from the actual situation. In the new configuration with the virtual train, this information will always stay synchronous.

You have many different trains, therefore you have to make sure the architecture is compatible with different generations of systems.

Applications of the virtual train: digital sinalling, real time monitoring, dynamical boarding information, the on-board situation etc. The virtual train is the backend that has links with many other applications. Trains have the content management systems, the intelligent train systems, intelligent maintenance system, many sensors. All these links are being translated to the back end through a hub. This hub standardizes all information to make all generations of trains compatible.

These systems are on Wagon level

G. Total IT architecture of Dutch case study



H. Slides expert review

Slide 1

Expert review

DIGITAL TWIN IN RAIL FREIGHT OPERATIONS

Slide 2

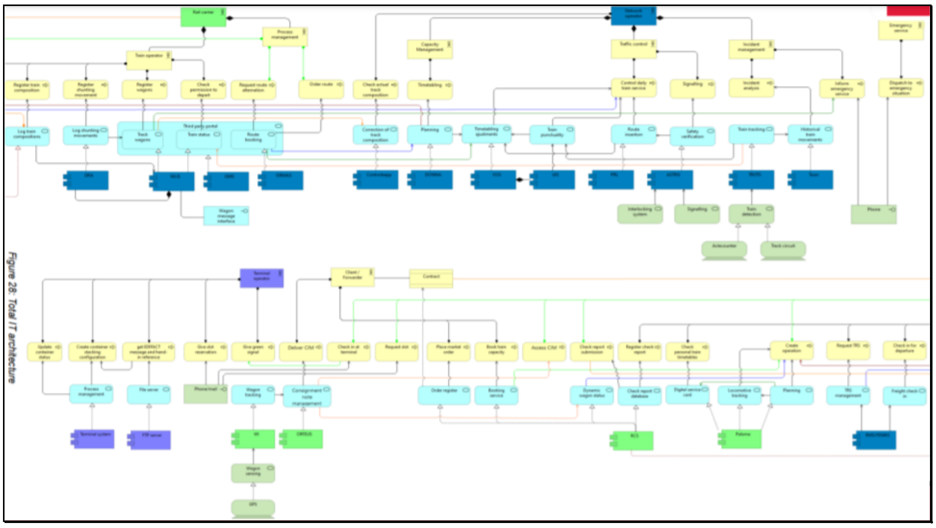
Het onderzoek

Veel digital twins op asset niveau, nog niet op operationionele niveau

Veel klachten op het gebied van transparantie omtrent goederenwagens

De vraag: Hoe zou een digital twin een oplossing kunnen bieden op transparantie gerelateerde problemen?

Slide 3



Slide 4

Totale systeem architectuur

→ IT aan processen aan stakeholders

Emergency services

Network operator
- Capacity management
- Traffic control
- Incident management

Carrier

Client

Terminal operator

Slide 5

Probleemstelling getrianguleerd

Improve **transparency** with an emphasis on **NCBG** areas by designing a MVP **2D digital twin** that satisfies business requirements of chain parties and innovation growth prospects while maintaining data security in order to **improve capacity management in freight train service and risk assessments in case of emergencies**

Slide 6

4 belangrijke stakeholder goals

1. As Capacity Management (network operator) I want to see shunting behaviour of my clients (i.e. the carriers) in order to increase the number of fulfilled capacity requests in my timetable.
2. As Incident Management (network operator) I want to see accurate wagon information in order to increase the effectiveness of risk analyses.
3. As Traffic Control (network operator) I want to see actual train positions in order to allow more ad-hoc capacity requests
4. As the Emergency Services I want more accurate wagon information in order to respond more accordingly to specific situations.

Slide 7

Architectuur

6 lagen (+1)

Te maken met sensors

Te maken met derden

MVC, Rest en OPC voor modulariteit

→ het moet mogelijk zijn gradueel te innoveren

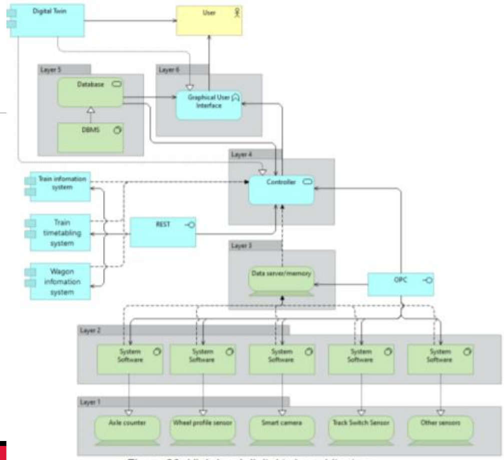


Figure 30: High-level digital twin architecture

Slide 8

Objecten

Data benodigdheden adhv van interviews

Vooral focus op de logica om een unit tabel en een composition tabel

Event tabel wordt nader besproken

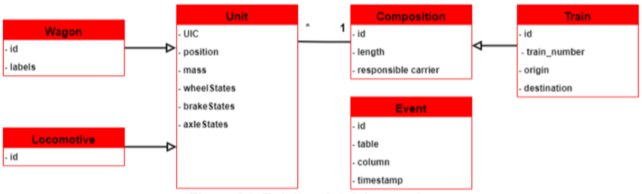


Figure 31: Relational database scheme

Slide 9

Event-driven architectuur

Data is asynchroon

Sensors genereren heel veel data

Er is een memory nodig als buffer

Events worden op een snapshot gemuteerd

Data-sourcing?

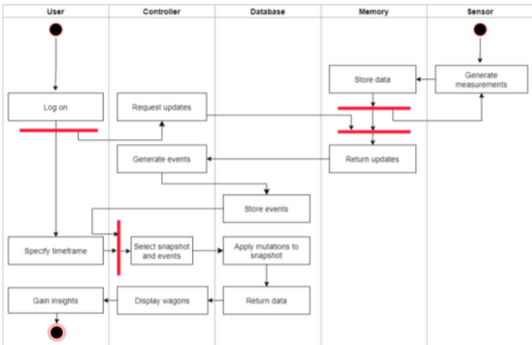


Figure 32: Activity diagram of event storing architecture

I. Evaluation IT Architect I

Occupancy	IT architect
Date	17-09-2021
Duration	60 minutes
Setting	Microsoft Teams

High-level architecture (Figure 7.1)

I don't really understand what the digital twin block means? Maybe this block is a bit redundant if it only serves as to show what the specific user application is composed of. To me it does not add extra information, but rather makes it a bit more intricate.

1. **Maybe this block is useless. In principle the full architecture depicts the digital twin**

Maybe it would be more straightforward to make the controller into a function rather than a service. But that is style, it all depends on how you have written it out.

2. **Noted. Agreed that they should both depict the same type. A controller is not a service.**

Now I look at it, I understand that the code of the controller and the GUI is in the digital twin. Maybe you might be right about adding the component. But I would make the View and the Controller both the same type.

3. **Still, the digital twin block indicates that only the GUI and the controller are twins. This is not true. This statement seems to be to not offend the designer**

But when I look at the layers, this is actually quite logical. If you look at the virtual train, that is practically also a digital twin, but regarding passenger freight. If you look at the technique: it makes use of a large memory grid. This is integrated with other parties. And if I compare you solution: when I look at layer 3, that practically is this memory grid. On top of that, there are many adapters to communicate with different devices. You specifically have OPC, but I do not think that is very existent in the virtual train. But the Virtual train does use many REST connections.

4. **The subject relates the architecture to existing solutions to confirm that the model seems logical. However, OPC might be too specific. Layer 3 seems to have a valid purpose.**

But the reason of that is that the virtual train is connected to many back-end systems, and not so much directly to the sensors themselves. These backend services have these connections, but the system itself not so much. That is why REST is so important to have.

5. **Shows that REST is useful for gathering system information. Does not add too much to the validation.**

On top of all these sub-systems you have many visualisations, and therefore, when I look at your system I see many similarities. It is quite in line with the practice.

6. **Layer 4, 5 and 6 also seem to be confirmed by the practice of the virtual train application**

Another example: I have done quite some projects with IoT. And that is also very similar. You have all these devices that read information from many objects, and these devices have system software. And then you have an intermediate hub, because you have many different protocols. And you had to convert some protocols to another. And this looks like layer 3. And there is still you raw structure of the data, which has to be prepared. That would be layer 4. That is the controller, and that forwards the data to the database in the right format. And then you could view it.

7. **Layers 3 and 4 is once again confirmed by the practice, another example.**

Database scheme (Figure 7.2)

A train is always a composition, but not the other way around. I understand that choice. You may have to explain very well in your report what exactly is a composition. But, yeah, logical!

8. **Confirms the composition to train relationship**

You only have to bear in mind that these fields change, right? So what happens when a composition becomes a train? How do you show that it is the same composition? But I think that will be made clear in your next model, my guess it has something to do with the event-table. But overall, the full model seems logical.

9. **The subject lacks the right information to judge the event structure at this point. But the subject does confirm that the choices made are well defined.**

Regarding the event table, the chronology is very important. Is the ID the sensor ID? Because if the id is of the event itself, it might be useful to add a "previous_event_id" field of some sort, to check the correct order of events.

10. **At this point the subject is informed of the event mechanisms. This is a valid statement: this would add to the chronology. This sounds like a blockchain setup?**

Event-drive architecture (Figure 7.3)

Okay, this needs some extra explanation, let's go over it slowly.

11. **The subject thinks of the model as intricate at the first glance**

But to me it is unclear what the difference is between an event and the saved data? Because often we speak as saved sensor data as an event. When something happens to the sensors, it stores the event. So what do you mean with event? When the temperature exceeds a certain level, for instance?

12. **There seems to be some confusion about the "event" naming.**

Ah okey, I am starting to understand. Maybe there should be an if statement for the case that there are no updates? Otherwise the system is stuck!

13. **Rightful observation, indeed the system would be stuck if programmed like this**

The snapshot explanation is sound, and indeed would decrease processing time. But then what do you mean with mutations?

14. **Again, there seems to be some confusion with "events" and "mutations". The snapshot idea has been confirmed.**

In theory, I have read a lot about event-sourcing. Once again, this model is very logical. The application landscape of the Network Operator is very event-driven. However, the events in that landscape are more on a triggering basis: if a train is delayed, a set of services will be told to execute certain tasks. And the order of events is in that case very important, and I think for your interpretation of an event as well. I think the chronology of these events is very important. So, I understand you have a scope to deal with in your research, but this is of importance.

15. **The subject relates to the Network Operator, and confirms using an example that the idea of event-sourcing is logical. The subject starts about triggering as if the current model does not incorporate that, the model might give the wrong impressions.**

There is a difference between how to handle streams and how to handle events. With streams you only provide the ID of a sensor. With events, you send all the information along with the ID. And you have incorporated that quite in your data model.

16. **First hints on the proper use of the "event" table**

The question is how the sensors provide data. Is that using timestamps? Otherwise, how would you know the order of data, and how would the memory know what data has not been sent yet, so the updates? I think you have to state very clearly that you made an assumption here.

17. **This looks like a limit of the research**

I start to really understand you event table and your event concept, but this might need some thorough definition.

18. **The subject had struggles understanding the "event" concepts, this should be renamed!**

Overall questions

The systems are logical, I think I have used that word enough haha!

19. **For most parts, the subject totally agreed to the designs.**

I think because you took some reference architectures, this system would definitely be feasible. It is existent, but what knowledge you added is the railway context. But if the Proof of Concepts are positive, I would not see why not. Especially that it is focussed on NCBG makes the system is interesting.

20. **Nice observation: the system should be feasible because similar architectures have already been realized.**

I think you took the core, you have thought about scaling. And that is per definition mvp. You have designed an essential piece. All other aspects will come in the future.

21. **The system is MVP proof, again reinforcing modularity**

I think your story is solid! Two hints: explain some things very well in your thesis! It is an intricate subject. And, translating the raw data to your event-structure. You show a table with a table and a column, and maybe you should keep it more generic. Events are in a chain, and that is currently not possible in this table. But it is a matter of how well you describe it!

22. **The event storage is useful, but might be more generic.**
23. **Overall observation: the subject was extremely positive about many aspects. The fact the subject really understood the modelling languages helps reshaping the models to be understood better.**

J. Evaluation IT Architect II

Occupancy	IT architect
Date	20-09-2021
Duration	45 minutes
Setting	Microsoft Teams

High-level architecture (Figure 7.1)

I see you use REST for getting information from other systems, but that is more of a pull-like approach that is not really used at the Network Operator. They publish data through JMS, that is by the principle of queues and topics. These are event-like messaging systems, with subscription mechanisms. Maybe, if you do not want to focus on one programming language, you could add some sort of "messaging" block, besides REST. REST is not necessarily too specific, but it is just a different mechanism then with the messaging systems. You have to bear in mind that with messages, the system has to keep track of the data itself. With rest, you can ask for the data anytime, that is different. So you might need additional databases. Maybe instead of REST, you could make it "messaging method" or something?

1. **REST seems to be too specific. Just like OPC as found in Appendix I note 4. There should be more generalizability. The model does not incorporate events from other systems.**

I do really see you have designed it to be focussed on modularity. In the principle it looks very logical.

2. **The subject mostly agrees with the architecture. Especially modularity is coming back, once again.**

If you would take an event-sourcing architecture, I would have the sensors publish events, and add to level 3 a standardisation of events. If you would add middleware for event-sourcing, such as Microsoft Azure Servicebus, or IBM MQ, or TIBCO EMS of the Virtual Train, those are all systems that work by the queues and topics mechanisms. It is the middleware that gets all of the events, and send them to all subscribed systems. That is asynchronous, and works using queues. So the servicebus will be inbetween layer 3 and 4.

3. **This statement came after showing the activity diagram (Figure 7.3). This is a very solid observation. Event-sourcing has not been incorporated into the design. But: there is middleware to implement, this could simply be added.**

Database scheme (Figure 7.2)

Noting to mention, looks good to me. One question: is the difference between wagon and locomotive very important? I would change it to one entity with a "type" field. Because it is normally no use to create an empty table. But that depends on whether this just shows relationships, or the exact database scheme.

4. **The overall architecture seems well. The empty "locomotive" empty caught attention, but this is merely to show object relationship. Developers could implement it how they interpret it.**

The event-table seems to miss a "value"-field?

5. **Correct observation.**

Service busses do not store events, you have to do that yourself. So the table is useful, but the naming is quite confusing.

6. **Again, the subject is confused by the naming of "events". This should definitely get some thought.**

Event-drive architecture (Figure 7.3)

Ehm, first of all the question: did you have to design this yourself or is that your choice? Because in principle the event-sourcing architecture is done by middleware. There is software for that. Generally speaking, you won't build this by yourself.

7. **The subject was very critical on this model. Having standard middleware is useful.**

The feeling you're giving me here, is that this model is way more complex then necessary. I mean, when you showed me the architecture, I did not expect this one of a controller.

8. **The model is too complex, and does not regard off-the-shelf solutions.**

I don't really understand how the login of the user is connected to the architecture? The manual data acquisition does not seem like event sourcing. Will this be periodically? What is the idea behind all of this? I seems like it is continuous, as if the controller is continuously gathering and shaping data. And you call that events? It does no really seem like an event-driven architecture… Event-driven would be if one system publishes events, and the other system consumes them and processes functions after a trigger.

9. **The subject remains very critical. It does not reflect event-sourcing. This is a valid observation, and should be incorporated in the redesign. The subject does agree that event-sourcing is the way to go!**

So instead of the sensors and memories being requested by the controller, it would be more logical to use the service bus as a trigger. Besides, I wouldn't name them events. Rather "mutations". You generate mutations, based on events, and the database applies those mutations.

10. **In the model, the controller should not be the trigger! The sensor and memory are unnecessary: they should be replaced by the service bus!**

The snapshot is useful, I agree with that one. Because you also want to see the mutations, so that would be possible.

11. **The snapshot mechanism has been agreed up on**

So the first part is not the way, but the last part is logical. You could use it after a few alterations.

12. **The subject was overwhelmed by the intricateness of the model, but after some alterations, the overall model was clear and logical.**

Overall questions
I think it is feasible. Due to your modularity you could go gradually. Step-by-step. Start with one sensor. It is MVP, the service bus was forgotten, but mostly it is the basics. 13. **The subject emphasizes modularity. It was even given as the reason of why it would be feasible.**

K. Evaluation IT Architect III

Occupancy	IT architect
Date	22-09-2021
Duration	30 minutes
Setting	Microsoft Teams

High-level architecture (Figure 7.1)

Generally, this seems quite logical. In itself it looks good. The fact that you take all of the sensors, and convert them to a central and general type, that is a good idea. We do that ourselves in Spoorbezetting. All requests are converted to a single format.

1. **The subject provides another practice example of layer 3**

But it looks in this way that it is real-time? I think you are missing a piece, the service bus. The Network Operator is very keen on this event-like system. That might be something that is missing here.

2. **The subject mentions the lack of events in the architecture. This is in line with IT architect II (Appendix J note 3).**

Most systems at the Network Operator put the messages in the queue. And these subscribed systems could process the events. You cannot save these events. You read from the queue and they are deleted from the memory. You should save them in the database.

3. **The subject confirms that REST is not the only connection that is important. More event-like systems exist. These could be added to Layer 2 perhaps? At least there should be a connection with the newly added service bus!**

You do have to bear in mind that, as you suggested, these sensors produce loads of information. Scalability in processing power might also be incorporated. Think of multi-threading, or docker containers.

4. **Very interesting thought! Modularity has been a buzzword, but scalability is at least as important! Especially with gradual innovations in mind.**

Database scheme (Figure 7.2)

It looks good, nothing wrong with it. Maybe you could merge the fields of unit and locomotive. And maybe also the wagon, so you add a “type” field. Besides, the UIC often show already whether a unit is a loc or a wagon.

5. **The subject agrees on the overall design. However, just like Appendix J note 4, the subject is focussed on the empty entity. But could stick to the design due to object relationship desplays.**

Also, you indicate correctly that a composition could have more then one loc, but we often also like to add whether the loc adds traction.

6. **This is an interesting find, but has not been requested by any of the stakeholders.**

Small remark: you said that I cannot judge about the fields. But trains have a date. A train number is unique for each day. So it could be the case that when a train departs at 23:00, it has the same train number as a train departing next morning at 7:00.

7. **This is a valid point, and a fact. This should be incorporated.**

Event-drive architecture (Figure 7.3)

Ehm, small detail about the "weekly" snapshot, I think that could easily be daily. That is generally the case at prorail.

8. **Good to know, but is not specified in the designs.**

I wonder whether it is logical to continuously ask for updates. You have loads of data incoming, and your model indicates that the controller has the initiative. If you wait for processing, than the controller has much to catch up to. A large backlog.

9. **The subject pinpoints the event-driven architecture is lacking! Again. The controller should not have the initiative.**

I think that your event-hub solution suits better. With the service bus. It would make sense that the servicebus takes initiative.

10. **The subject agrees with IT architect II, a service bus is a need.**

Overall questions

Overall, I think the complexity would lie in the different sensors. They all provide different data formats. And you try to have that centralized, which is good, but it is not that easy I think. But this is something you could have done by different teams, implementing such an architecture could be done by putting different teams to different aspects of the system. They are independent.

11. **This subject is somewhat more critical on feasibility. There is no doubt the design is feasible, but layer 3 might be more intricate than expected.**

I started talking about the Docker containers, I am really a fan of them. Definitely with these types of architectures, because it improves scalability. You could allocate different tasks between the event processing. Or, you could allocate the events between different dockers of the same processing feature. Maybe docker different types of processing techniques for different events. By using docker, you could perform tasks simultaneously. Or faster one after the other, in serial.

12. **These arguments are very interesting. A new aspect of scalability. This is very relevant for the design, but has not been incorporated. Future research!**

L. Evaluation Capacity management

Occupancy	Capacity management (Network Operator)
Date	16-08-2021
Duration	45 minutes
Setting	Microsoft Teams

Introduction

This was a very enthusiastic and talkative interviewee. The subject is active in rail freight as a logistics expert, and is focussed on issues regarding capacity. The subject emphasizes that he/she had no experience in passenger traffic. Further, the subject mentioned that there are tight collaborations with traffic control.

Function of capacity management and traffic control

We are busy with all sorts of issues with the focus on capacity management. The team is divided between the areas that we focus on. Of each of these track sections, or sometimes routes, are known very well by each of the subteams. These subteams then analyse their respective sections for bottlenecks. You basically want to find out how you can increase the capacity on your designated area.

There is the ambition to increase the traffic regarding rail freight by over 40% by 2030. Our task is to map the bottlenecks that would withhold these developments. It could be that you have too little tracks in a specific section, or that the tracks are maybe not long enough. The latter could be relevant to the amount of 740 meter trains they are planning to increase.

In that manner you are going to argue: for what purposes do we need certain track sections? These bottlenecks arise from analyses on the basis of pure data, and performing calculations to this dataset.

1. **This was a first hint on the need for data analysis, rather than visualisation**

This is done by gathering information about a specific track section, and by analysing what trains have been installed there, and for how long. A little note: this is all regarding CBG. That is where we have the required data for, and that is something we can map quite clearly. Even though CBG is transparent, analysis is tough.

2. **It does show that our finding that current solutions are not available to NCBG. This supports the design problem**

For us capacity manager it is of importance to know how a certain area functions and what to expect of it. In that way it is best found out where the bottlenecks arise. My designated area is “Lage zwaluwe”, which is a main junction for the Botlek area.

3. **So the subject’s knowledge is about the right areas involved in the scenario**

I have to know everything that happens on that area, so I need to see from all traffic, where is it coming from, where is it going, how long is it installed at the site, etc.

4. **Hinting on the need for a need for "origin", "destination" and the playback feature**

In this way you map a certain area before you perform analyses. And then you try to determine if the data is in line with you expectancies.

5. **maybe is the visual interface an aid?**

This is all long-term, maybe 2 or 10 years forward. The short term matter is more a question of either capacity allocation or traffic control. They are more focussed on, what capacity could be planned in the upcoming moments? This is more of an ad-hoc matter. They are more focussed on: if there are ad-hoc requests, is it possible to honour these? And why, or why not? They would benefit more from some kind of visualisation, as they would like to know some as-is situations. We are more focussed on the insights data gives us, there is an important difference.

6. **Perhaps the interface is more of use for traffic control, and not so much for capacity management**

TRSs on the NCBG are not ad-hoc either: they are allocated a year beforehand. There are discussions for when there are conflicting desires. In these discussions, there is input from the Clients with conflicting desires and together it is discussed how the conflicts can be solved.

After the allocation of TRSs, there is always an option to make last minute requests, but these can only be honoured if a TRS has not been allocated yet.

7. **Would this maybe be improved by having such a visualisation?**

Evaluation (from now on the artifact will be shown to support the discussion)

First of all, for the greatest part, we do not really need a visualisation for the track occupation levels.

8. **Very clear. The subject does not directly sound optimistic about the shown GUI. Now why is that?**

We have something similar (TOON) and we use that piece of software to replay some shunting movements that we don't really understand from pure data. This software shows us how a train has driven, which is useful if we don't understand how a train has moved, or when its path seems illogical. We can really play that back as well. But, this is only for CBG.

9. **So, TOON is used by this party it seems. So this would indicate that a similar system would also be used when applied to NCBG**

However, what you are showing me would be very useful for Botlek or Moerdijk: that is mostly NCBG.

10. **Previous note is confirmed, a system would be used in NCBG**

We do not have such views for NCBG, there is no TOON. But in these cases, both for me and for traffic control, there is only the question of track occupancy. The geographical interface might be over the top and some low level map could be more clear.

11. **The subject remains doubtful, and expresses indirect disagreement with the used view**

The software you are showing me might be more applicable to the roles of Traffic control. They are concerned about the now, the current positions of trains, and foremost the visualisation of it.

12. **The role of capacity management may be mis understood. The system does show potential in the given scenario, which seems to be more applicable to traffic control**

There is one scenario that I can think of that a visualisation could be useful. When we are in a discussion with the Clients in case of conflicts, the visualisation could provide an aid for discussion when we need to solve these conflicts. For example when there are some inefficient shunting movements, we could discuss together based on the view how shunting behaviour could be altered in order to honour more requests.

13. **It still remains unclear how specifically this would benefit, but it sounds like the cost-benefit is not favourable**

Now that I think of it, how Traffic control and capacity management could both get help from a digital twin is by showing delayed trains. If both parties would collaborate a bit more and use the digital twin, some TRS allocations could be performed on a more ad hoc basis. This is already possible on CBG, but not yet on NCBG.

14. **This has not been incorporated clearly: it would be possible to view trains and decide yourself if a train is delayed, but there are no ETA's**

The passing by train that you showed might be useful, but most important to me would be: what is left behind? So the fact that you could select time windows and play back what happened on wagon level is actually quite useful. If this would be visual I could decide better if the area could benefit from extra tracks. Imagine that a train had to shunt just because the tracks were too short. These steps would be unnecessary if the tracks were extended.

15. **This does confirm requirements 4.4, 4.5, 4.6, 4.7 and 4.8. Also a quite useful explanation was delivered**

But, overall what you are showing me here is mostly a traffic control tissue. But yeah, if you would see this scenario on NCBG, it would be possible to make the trains occupy both TRSs.

16. **carefully confirms the scenario being useful to the Traffic control case. This provides some validity, but the case is in no way related to capacity management**

But: cost wise? What you are doing here is practically transforming NCBG into CBG, and the costs of smaller proof of concepts that try to do just that might exceed the 75 million per PoC. So it is a question of "is it worth the investment?". And sometimes it is, but other times it is just not that interesting.

17. **Important to knowledge question: seems not feasible in this case. But it does add to the modularity requirement → should come gradually**

For me it's more at the wagon level, although with traffic control they would really like to know the exact train composition.

18. **confirms requirement 4.4 again. Seems to reject 4.5 however, which contradicts with earlier statement. It seems to confirm requirement 4.5 for traffic control, which has not been incorporated → false negative**

I just want to know how long the train is, where it's going, where it's coming from.

And which Carrier is involved. Especially if you were still working with trs at all, which I doubt. Because without ncbg it's just putting in routes. I just need to know the routing. I see speed as a nice to have. This is especially useful for when a train is going somewhere. At moerdijk you are allowed 40, so the question is how fast they actually drive. So could they be faster?

19. **Very useful section. Of requirement 2 the following fields are confirmed:**
    - Train: origin
    - Train: destination
    - Composition: responsible Carrier
    - Composition: length
    - Composition: speed

We simply collect a year's worth of data. So visually this is not possible. So then you need a dashboard. But the visualization would be useful no matter how I use it now. And then you want time-specific. This is already possible with TOON.

20. **The subject remains carefully doubtful in its intonation. Seems to not want to disappoint the researcher, and therefore the use for such a visualisation seems to be unworthy of its investments.**
21. **Conclusion of the interview: traffic control would like to have this feature, but to capacity management the data of layers 1-5 are most important.**

M. Evaluation Consultant IV

Occupancy	IT Consultant capacity management
Date	24-08-2021
Duration	25 minutes
Setting	Microsoft Teams

Introduction

This was a rather informal meeting; the subject was short of speech, but gave some interesting insights that seemed in line with the evaluation of appendix L. The interview became less of an evaluation of the specific requirements, but rather felt like a discussion as to why the twin could be useful. The interview could be used to back-up statements in the other evaluations. This subject has been actively involved in the Botlek PoC, which provides knowledge to link the digital twin to the use cases of Botlek. Most of the interview turned towards the botlek case, but some links towards traffic control were made. The subject insisted and started a demo of the dashboard of the Botlek, which was followed by the demo of the DT.

Botlek demo

There was this feeling at capacity management that TRSs were allocated too broadly. This PoC uses the data of TROTS and the planning systems supported by 8 sensors on rails (these are axlecounters). On this way we could see what trains were driving somewhere, how long an operation took place and what route a train drove.

In short: the hypothesis tends to be true, we indeed see that TRSs are reserved for a too long time, more then necessary. This hints on a cultural issue: Carriers do not want to plan fewer time in, because they are afraid to lose their slots.

The dashboard is mostly of data visualisations, and there are different tabs with different kinds of visualisations. Knowing whether a train has left a TRS is already possible, just not visually or real-time. There is no geographical real-time view, capacity management is mostly concerned with data analysis, where traffic control is more interested in the former.

1. **General conclusion from the demo: still traffic control seems to benefit from these views on NCBG. But once again, capacity management is very focussed on data analysis, rather than geographical locations.**

DT demo

The scenario is relevant to me, as it concerns a problem that is existent in the Botlek.

2. **Confirms validity of the scenario**

In principle there is currently already the option to see the problem of this scenario, but it is always afterwards. It is purely analytical. Real-time controlling the traffic in NCBG is not possible as of now. Currently the trains have to wait, and that is what the PoC was about. 1: the Network Operator wants to increase capacity, and 2: they do not want to make the area NCBG because that is too expensive.

3. **the current solution is not feasible, but this emphasizes the modularity requirement**

This visualisation could really help with playing what has happened. For instance with a group of representatives you could have table meetings to play back what has happened on the area. At certain times you can really pinpoint what the bottlenecks are. I think that a visualisation really helps.

4. **this supports Appendix L note 15**

With these table meetings it would be nice to have different parties within the Network Operator to join. The different functions could be nicely integrated through such a central system. All of the business functions have different data sources that are shared too little. If they could look at the same set of data, you make some processes more effective. If one party could warn the other party about delayed traffic, some TRSs could be reallocated. This is the main focus at Botlek.

5. **This nicely reconfirms Appendix L note 14. It seems that this would be useful to both capacity management and traffic control.**

The table meetings with the Carriers could benefit planning-wise. The planning is full, but we have read from the data that TRSs are allocated way too broad. This would help solving the cultural issue that exists currently, in which Carriers greedily ask for more time than they need, just so that they have more flexibility. By having meetings about shunting behaviour, you could maybe get them to shunt more efficiently, and to reduce their TRS time.

6. **This is backing up the shunting behaviour notes from Appendix L. This subject even calls it boldly a “cultural issue”.**

However in terms of technological advancements, I think the application you designed comes too early.

7. **See note 3**

I see another advantage of having this visualisation: you can imagine that not all data is complete when you have many involved data sources. The sensors are accurate, but to a degree. And sometimes it could happen that data is presenting strange situations, that can not be understood. And having a visualisation to playback what happened, helps understanding the problems in the data.

8. **This is a new mentioned advantage of the system, it could be a useful tool in this way**

Full integration is utopian, I believe. But it is desired.

9. **Again hinting towards modularity (Appendix L note 17)**

The thing is: making all areas CBG is just too costly, especially compared to what benefits are achieved. Therefore, not all of the areas are fitted with sensors. Would it be possible to have these areas added one by one, instead of having all together? It would be nice if you could add some sensors and areas through some buttons.

10. **Perfectly confirms requirement 1.3. This is a nice to have requirement for system administrators, a stakeholder group that has not been incorporated, but is of course really important.**
11. **Final note: this was definitely a different interview than expected, but definitely not useless. It could serve very well as a supporting evaluation of Appendix L. Some nice new perspectives were added. Once again it showed that the subject really had to be creative to find advantages: the fact that the advantages were not that straightforward and came with some thought implies that Capacity management would not gain too much value of such a visualisation, but Traffic control might.**

N. Evaluation Consultant V

Occupancy	IT Consultant Incident management
Date	09-09-2021
Duration	35 minutes
Setting	Microsoft Teams

Introduction

This consultant was very clear in giving feedback, the subject did not hesitate in pointing out what was not liked. The history is this subject is widespread, but its experience in the logistics field is emphatical. The subject is well known in the IT world, with its knowledge starting as a student. Currently the subject is a data analyst and an architect for WLIS. This gives the exact knowledge field to judge the digital twin.

Demo of DT

Okay, firstly, when there is a fire they first want to know: "what freight is involved?", "what wagons are involved?", "in what order?", "are there any hazards?". In rail freight you have a lot of toxic and inflammable substances and it is important to know what is positioned where. I think that is currently displayed, and that seems right.

1. **Confirms requirements 4.3, 4.5 and 4.11 for the Emergency Services**

And this is not only on real-time information, they also want to playback what happened in case of catastrophes, "what was where?".

2. **Confirms both requirements 4.3 and 4.4, 4.5 and 4.8 for incident management**

I think what you are showing me is mostly automated data generation? You might have to bear in mind that sensors have a certain margin of error. So what happens when something goes wrong with a car that was never registered? I think there might be some serious liability issues. So the question with regards to that is, what becomes the role of the Carrier? If the Carrier enters a shunting movement we speak of a mutation, but these automated measurements are continuous. So what happens with the liability of the Carrier? Is he still accountable?

3. **This adds to requirement 1.1, there should be a serious note to the requirement**

And I also do not think that you should work with a system that makes suggestions, because then you would have lazy Carriers just accepting every movement the system proposes. Maybe it is best to have the Carriers input cross-checked by the system. It could warn them if the input is not what was expected.

4. **These findings are very thoughtful and are in line with own findings in Section 6.1.6. Therefore requirement 1.1 should be reassessed.**

See, what happens when the fire department arrives: they just start counting wagons. WLIS shows them that the sixth car is a liability. They just count, 1, 2, 3, 4, 5, okey it is that wagon. And when the information in WLIS is wrongful, counting just is completely senseless. This is where the automated identification of wagons could support the improvement of risk assessments. The reason that these wagons have been entered wrongly so often, is due to the Carriers get a very low margin on their business. They have little time, and they feel in no way happy about thoroughly checking whether every movement is entered correctly. They just enter it in the systems because they have to. And you could use such a digital twin to correct them, in order to increase data accuracy. Carriers would have an incentive to cooperate, because everything will be cross-checked. It has to be an addition.

5. **Even though requirement 1.1 should be reassessed, the need for automated tracking is recognized. Partially satisfied.**

When the Emergency Services have been dispatched, they are provided with WLIS reports.

I think this is the total weight of the wagon? I think that should be the weight of the contents, that is more interesting. That is something that risk analysts would be interested in.

6. **Shows that some changes need to be made in Figure 7.2**

I think there is also a bit of an informational overload, with all of these wheels and axles. Let me show you how WLIS works.

7. **Information overload is a term that seems to be more existent in this interview.**

Demo of WLIS compared to the DT

For WLIS there has already been done a lot of requirements engineering. You have a list of wagons, not a geographical view but rather a list. The wagons in the list are clearly displayed with labels and wagonnumbers. These labels are standardized, and just showing the labels is plenty. Besides, a wagon could have more than one content, and showing labels provides both ease and privacy. Also there are different wagon statuses: “loaded”, “unloaded” and “cleaned”. In the first two, the wagons need to have labels because of damps that might still be in the containers. Having labels is different to having contents.

8. **Some aspects of Figure 7.2 need to be altered. Having the “contents” field both contradicts with requirement 2.1 and adds to informational overload. Adding labels provides clarity and privacy.**

Two important comments on your system: wagon numbers are important, and the responsible Carriers are important. You already have that, but in your system you first have to click. That are too many actions for showing important information.

9. **Shows requirement satisfaction of the fields UIC and responsible Carrier. But again ease of use is emphasized.**

So instead of your alerting triangles, the labels are way more important. You need more of a symbolic visualisation, rather than this correct visualisation. Maybe you could add some colour coding. But all these wheels you have, that is just too much.

10. **Informational overload, again. More focus on ease of use. It remains the question whether the asset health fields should be incorporated for the use of risk warnings**

It is important to maintain clarity. All necessary information has to be shown quickly. I think that is the strength of a digital twin, that everyone has the same information very quickly. But! You have a matter of security! Competitors may know nothing about the wagons, except for labels and generic data. So it must be assessed thoroughly what information is essential.

11. **Confirms the benefit “COP”. Furthermore confirms requirements 2.1 and 4.1**

I think the basis is that the entire sector shares the same set of data. That would provide synchronous transparency for all involved.

12. **Also came forward in the other evaluation interviews: COP is a main benefit.**

Regarding the scenario, I think it is useful that you merged these two goals (2 and 4) as one. These stakeholder share the same set of goals.

13. **Adds some validity to the scenario**

O. Full requirement acceptance table

Requirement	Requirement level
R1. The system shall support a 6-layered, modular architecture.	Domain
R1.1 The system shall allow gradual innovation through low-effort extension of • The amount of sensors connected • The amount of RESTful sources	Product
R1.2 The system shall incorporate the REST and OPC principles	Product
R1.3 The system shall incorporate the MVC principle	Product
R1.4 The system shall implement an independent, automated wagon tracking and identification	Product
R1.5 The system shall implement an independent, automated asset health detection	Product
R1.6 The system shall have a log-in function	Product
R2. The system shall support a relational database scheme.	Domain
R2.1 The system shall incorporate compositions composed by one or more units	Product
R2.2 The system shall incorporate wagons and locomotives as units	Product
R2.3 The system shall have the option to transform a composition into a train	Product
R2.4 The system shall support the storing of events	Product
R2.5 The system shall never incorporate commercially sensitive data	Product
R2.6 The system shall incorporate the specific fields as given in *Figure 7.2*	Design
R3. The system shall support an event-driven architecture.	Domain
R3.1 The system shall check the memory for new sensor-data	Product
R3.2 The system shall generate events from incoming sensor data	Product
R3.3 The system shall incorporate a snapshot system for the database	Product
R3.4 The system shall apply events to the snapshot nearest to a user given time-interval	Product
R4. The system shall implement a GUI.	Product
R4.1 The system shall have a 2D geographically correct map of the rail network	Design

R4.2	The system shall have button for a real-time mode	Design
R4.3	The system shall have a button for a historical mode with a time-interval selection	Design
R4.4	The system shall display individual wagons on the rails	Design
R4.5	The system shall display multiple wagons as one composition/train on the rails	Design
R4.6	The system shall display the movement of wagons and compositions on the rails	Design
R4.7	The system shall have an option to play or pause a certain timeframe	Design
R4.8	The system shall display asset health of the individual wagons	Design
R4.9	The system shall display warnings for risks and hazardous materials	Design
R4.10	The system should display wagon information of a selected wagon	Design
R4.11	The system should display train/composition information of a selected composition	Design