Model-Based Systems Engineering Approach to an Intermodal Freight Terminal

A thesis submitted in partial fulfillment of the requirements for the degree
M.Sc. Mechanical Engineering
at the Department of Mechanical Engineering
of the Technical University of Munich.

Thesis Advisor Prof. Dr.-Ing. Johannes Fottner
Chair of Materials Handling, Material Flow, Logistics

Supervised by Maximilian Schöberl, M.Sc.
Chair of Materials Handling, Material Flow, Logistics
Marc Ehret, M.Sc.
German Aerospace Center

Submitted by David Wucherpfennig
Agnes-Bernauer-Str. 71, 80687 Munich
david.wucherpfennig@tum.de

Submitted on May 1, 2021, Munich

Inventory Nr. fml 2020101

Preface

Die vorliegende Arbeit entstand unter der wissenschaftlichen und inhaltlichen Anleitung von **Maximilian Schöberl**, M. Sc., wissenschaftlicher Mitarbeiter am Lehrstuhl für Fördertechnik Materialfluss Logistik (fml) der Technischen Universität München.

Vereinbarung zum Urheberrecht

Hiermit gestatte ich dem Lehrstuhl für Fördertechnik Materialfluss Logistik diese Studienarbeit bzw. Teile davon nach eigenem Ermessen an Dritte weiterzugeben, zu veröffentlichen oder anderweitig zu nutzen. Mein persönliches Urheberrecht ist über diese Regelung hinaus nicht beeinträchtigt. Eventuelle Geheimhaltungsvereinbarungen über den Inhalt der Arbeit zwischen mir bzw. dem Lehrstuhl für Fördertechnik Materialfluss Logistik und Dritten bleiben von dieser Vereinbarung unberührt.

Munich, May 1, 2021

Abstract

The complexity of intermodal freight terminals poses a challenge for logistics planning. This also applies to the Next Generation Train CARGO (NGT CARGO) logistics terminal, which is embedded as a future intermodal transshipment hub in a high-speed freight train concept developed by the German Aerospace Center (DLR). Previous studies revealed that the existing terminal concept is only partially capable of facilitating the required intralogistics processes.

This thesis investigates the successful application of Model-based Systems Engineering (MBSE) as guiding approach to a detailed system architecture of the NGT CARGO logistics terminal. Focus is laid on intralogistics freight handling. A tailored MBSE approach is derived and applied within a comprehensive architecture modeling process. As key outcome of the specification process, a product architecture variant is specifically designed for selected types of goods. The resulting variant is verified and validated from a logistics perspective. The underlying system model enables the creation of further NGT CARGO logistics terminal variants.

The results of this work demonstrate that MBSE is capable of successfully guiding the architecture development of complex logistics facilities. Experienced advantages and disadvantages of the application of MBSE are discussed. Integration of the promising MBSE approach into current and future logistics planning is proposed.

Contents

Acronyms and Symbols

Abbreviation	Description
AGV	Automated guided vehicle
CAD	Computer-aided design
CEP	Courier, express, and parcel
CRRC	China Railway Rolling Stock Corporation
DCRM	Distribution Center Reference Model
DRM	Design Research Methodology
DS-I/-II	Descriptive study I/II
DLR	German Aerospace Center
ERP	Enterprise resource planning
ERTMS	European Rail Traffic Management System
FAS	Functional Architectures for Systems Method
FIFO	First-in first-out
IDEF0	Integration Definition for Functional Modeling
INCOSE	International Council on Systems Engineering
ITS	Intralogistics transportation system
LDHV	Low density high value
LIFO	Last-in first-out
LS	Loading system
LU	Load unit

LU Info	Load unit information
MBSE	Model-based systems engineering
NGC	Next Generation Car
NGT CARGO	Next Generation Train CARGO
OMG	Object Management Group
OOSEM	Object-Oriented SE Method
OPM	Object-Process Methodology
PLM	Product lifecycle management
PS	Prescriptive study
PSC	Problem-solving cycle
RC	Research clarification
RFLP	Requirements-Functional-Logical-Physical Approach
RUP	Rational Unified Process
SE	Systems engineering
SOI	System-of-interest
SRS	Storage and retrieval system
SysML	Systems Modeling Language
SYSMOD	Systems Modeling Toolbox
TAI	Transshipment area inbound
TAO	Transshipment area outbound
TKT	Ton kilometers transported

TUM	Technical University of Munich
UC	Use case
ULD	Unit load device
UML	Unified Modeling Language
VDI	Association of German Engineers
WIA	Warehouse interface area
WH	Warehouse
WMS	Warehouse management system

1 Introduction

In the first chapter, an overview of the initial motivation for this thesis and the following research approach is given. Section 1.1 starts with a brief description of the initial situation and explains, why intermodal terminals play a major role in future freight transportation. Since the thesis bases on the Next Generation Train CARGO (NGT CARGO) project carried out by the German Aerospace Center (DLR), Section 1.2 introduces the NGT CARGO project and states the problem addressed by this thesis. The resulting research question and the defined research objectives are presented in Section 1.3, before the chapter is concluded by Section 1.4 describing the research approach and the structural framework of the thesis.

1.1 Initial Situation

Currently, the world is facing the COVID-19 pandemic crisis caused by a coronavirus [Wor-2020]. In regard of logistics transportation networks, the ongoing crisis revealed that the rail freight transportation is a crucial part of a reliable good supply. However, this conclusion is not as obvious as it seems to be, since rail freight transportation used to play a minor role compared to the road freight mode [Sch-2020].

Regarding the national freight transportation market in Germany, around 130 billion ton kilometers transported (TKT) were delivered by rail in 2019. This represents a total transportation market share of 19% [Wei-2020a]. Although a growth in total rail transportation performance is about to come, its market share is even expected to lower down to 18% (154 billion TKT) by 2030 [Bun-2016, p. 55]. Meanwhile, road freight traffic is estimated to cover around 73% (607 billion TKT) of all transportation service in 2030, as shown by Figure 1-1.

A core issue concerning these numbers is that the transportation sector is in charge of roughly 25% of the greenhouse gases emitted in the European Union [Eur-2020]. In 2019, the European Commission announced an action plan to become the first climate-neutral continent - the 'New Green Deal' [Eur-2019b]. In order to achieve the environmental goals

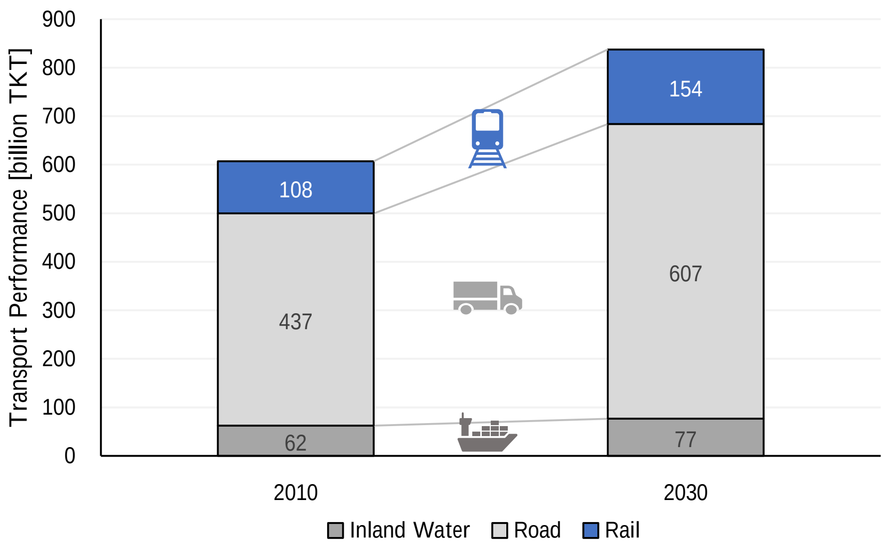


Figure 1-1: Forecast for development of transport performance in Germany with initial base year 2010; values in billion TKT; share of rail is highlighted [Bun-2016, p. 55]

for 2050 set by the European Commission within the New Green Deal, it is necessary to reduce emissions in the transportation sector significantly. As rail traffic produces only 0.5% of transportation-related greenhouse gas emissions, it is the most sustainable mode of transportation. Therefore, it plays a key role for the reduction of the CO_2 emissions in the transportation sector. An increase in the modal split of rail freight is stated as a major step on the way to reach the European Union's climate goals. To address this issue, the European Commission declared the year 2021 to be the *'European Year of Rail'* and launched a fourth railway package to revive the rail transportation system [Eur-2020; Fel-2021].

However, the recognition that a modal shift is necessary to facilitate a sustainable transportation system is not new. Already in 2011, the European Commission published a white paper, postulating among others the objective to achieve a 50% modal shift for distances over 300 km from road to other modes of transport, mainly rail, by 2050 [Eur-2011]. Concerning Germany, the master plan rail freight published by the German Federal Ministry of Transport and Digital Infrastructure in 2017 proposes several spheres of action and corresponding measures to support the change towards an efficient and sustainable freight rail transportation system [Bun-2017]. However, the latest numbers do not show any significant improvements in modal share [Wei-2020a].

To understand the issues rail transportation is facing a deeper dive is necessary. The main difficulties of the rail transportation network are inherent to the system of rail. Construction and maintenance of a broad rail network including transshipment terminals is very costly. Once built, tracks are fixed and limit the rail vehicles in route variability. Overtaking procedures are only possible if partial side tracks exist. National differences in track gauge, electrical system, or train control system complicate cross-border rail transportation[1]. Furthermore, the capacity of tracks is determined by physical, legal, or organizational factors, such as minimum train distance. Based in historical reasons, freight and passenger rail traffic largely shares a common track system within Europe. Since faster passenger trains often operate in a prioritized state, freight trains have to give way and use side tracks. Apart from a reduced average transportation speed, this procedure leads to a limit in the maximum train length of 740 m for freight trains [Mei-2013].

Due to these structural difficulties, rail transportation used to focus on the market of goods with high volume or weight such as mineral oil, automotive or bulk goods like coal. When transporting these goods, the possibility of efficient block train services from source to sink enabled the rail transportation to gain a major competitive advantage [Wei-2020a]. However, in times of a 'New Green Deal' and rising environmental issues, the demand for these goods is not about to grow. Instead, the future freight markets in Europe will be driven by so-called low density high value (LDHV) goods [Jac-2014]. These types of goods are characterized by being rather small, volatile in volume and they postulate top requirements like reliability, price, and time. Today, these requirements are met best by road transport, providing a highly flexible, cost effective and - regarding short distances - rapid transportation network [Isl-2018]. Examples for LDHV goods are parcels, foods and other perishable goods, or pharmaceuticals [Ehr-2020].

Due to the previously stated properties of rail track network and missing railway sidings, serving this market independently is difficult for rail transportation. Thus, an intermodal transportation system with train service on long haul and road vehicles as first and last mile service seems to be promising for realizing the modal shift towards rail especially for LDHV goods [Bun-2017, p. 27; Wei-2020a]. Intermodal terminals are identified as one of the critical elements of the transport chain to a successful implementation of intermodal freight transport for LDHV goods [Kil-2008; Isl-2018].

[1] Among others, this issue is addressed by the European Rail Traffic Management System (ERTMS) initiative [Eur-2012].

1.2 NGT CARGO Project and Resulting Problem

To address the EU´s vision to shift freight transport from road to rail, the DLR develops a comprehensive intermodal vehicle and logistics concept, the NGT CARGO. Its main goal is to support the modal shift of freight transport on the main run from road to rail by offering the possibility to transport LDHV goods on the rail network. The core of the project is a future highly automated high-speed freight train (Figure 1-2).

Figure 1-2: Digital rendering of the NGT CARGO train (DLR)

It consists of two automated train power units and ten rail cars in standard configuration. Due to its high-powered locomotives and aerodynamic design, the train is able to travel at an operational speed of 400 km/h. Being capable of direct train to train communication, the NGT CARGO can be virtually coupled with other high-speed trains (e.g. passenger variants from the NGT project). Applying this procedure, the limited rail track network can be used more efficiently and the operational flexibility of the NGT CARGO is increased [Win-2017a]. Regarding the operation concept itself, the NGT CARGO logistics project contains work on an operational scenario using a reference route from Madrid to Bucharest [Win-2017b; Mön-2020].

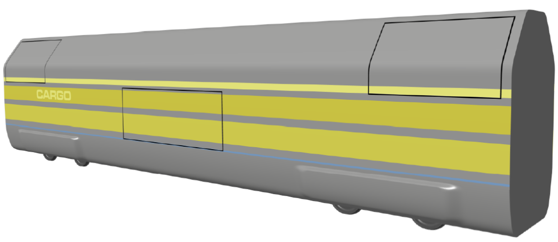


Figure 1-3: Digital rendering of an NGT CARGO railcar (DLR)

Figure 1-3 illustrates a single NGT CARGO railcar. The NGT CARGO railcars provide a double deck load room. It is accessible from both sides each through two doors on the upper level and one central door on the lower level. The railcar is designed to hold Unit Load Devices (lower deck only) or standard European pool pallets. Loading, unloading, and the movement of the transportation units within the rail cars is realized by roller floors with automated driven roller or chain system. To serve fine distribution towards logistics sidings apart from the head powered long haul route, the battery-powered rail cars are designed to operate autonomously and self-sufficiently on short distances [Böh-2019].

However, not every stakeholder in the LDHV market can rely on a logistics siding, which could be served by the NGT CARGO rail cars. Therefore, the NGT CARGO train is designed to operate in an intermodal transportation network. Thus, a logistics terminal is needed, where load is transshipped from the NGT CARGO to other means of transport and vice versa. Being the intermodal interface facilitating this transshipment, the logistics terminal is an essential part of the concept. Due to the high requirements regarding transport times, reliability, and costs linked to LDHV goods, an effective transshipment between the NGT CARGO vehicles and other means of transport is vital. In consequence, the logistics terminal has to satisfy these requirements to support the competitive advantages of the high-speed transportation concept [Böh-2019].

Based on previous work at the DLR, Figure 1-4 shows a futuristic, conceptual design study how the logistics terminal could look like. However, a system analysis of the logistics terminals investigating on system context, stakeholders, requirements, and top-level system

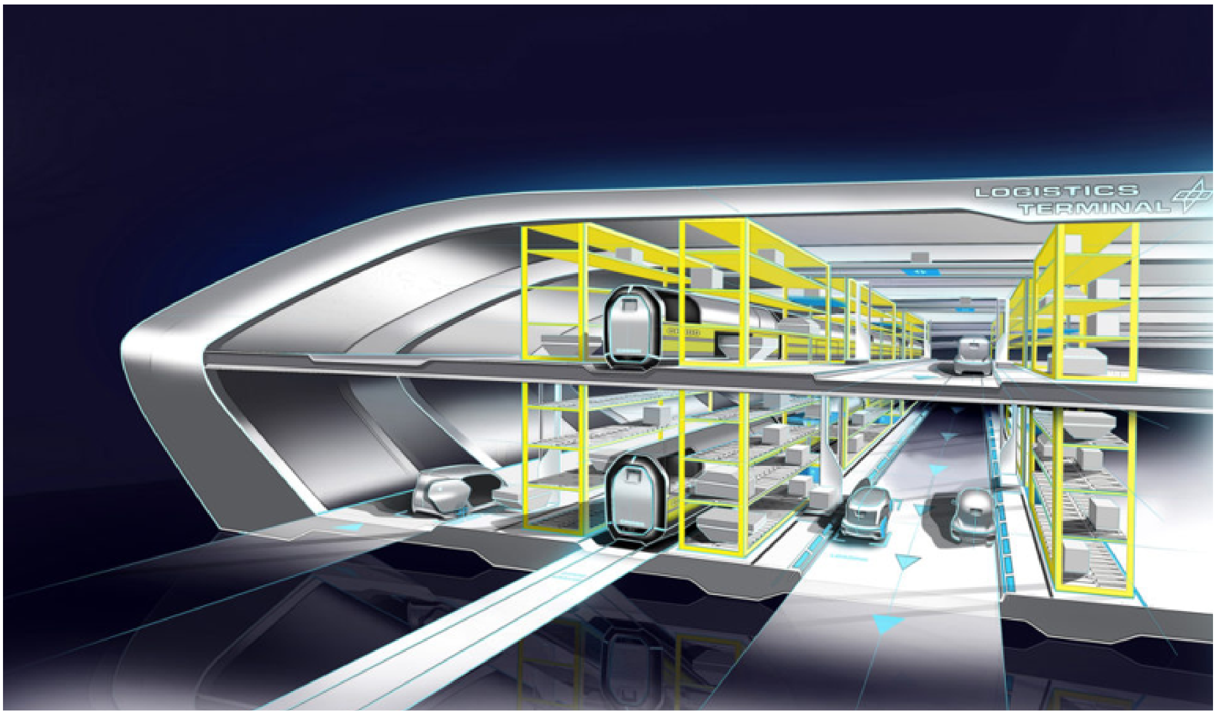


Figure 1-4: Digital rendering of the NGT CARGO logistics terminal (DLR)

processes has shown that the conceptual design is only partly matching with current logistics processes dealing with the transshipment of the targeted goods. Furthermore the study revealed, that system architecture is complex and strongly depends on the type of goods. Therefore, a more detailed and realistic system architecture of the terminal is necessary to address the various requirements of logistics stakeholders. This includes the specification of the intralogistics processes.

How should a system architecture for such a terminal designated for the transshipment of LDHV goods transported by a highly automated high-speed train be developed?

Various literature exists for planning and design of conventional intermodal terminals [Mül-1997; Bon-2000; Car-2008], or refer to specific conventional intermodal transshipment terminals [Wid-2017]. Conventional in this context means the transshipment of containers, swap bodies or semi-trailers and refers to low-speed freight rail service. According to *Ehret et al.* few concepts for transshipment terminals in context of high-speed freight transport exist. Yet, these concepts do not focus on intermodal transshipment of unit loads like pallets or ULD [Ehr-2020]. In his dissertation *Woxenius* proposes the usage of a systems approach when developing a small-scale intermodal terminal. This systems approach strongly relates to the basic principles of systems engineering (SE) with focus on actors, requirements and iterative system development cycles [Wox-1998].

According to *Friedenthal et al.*, SE is *'a multidisciplinary and holistic approach to develop solutions for complex engineering problems'* [Fri-2014, p. 16]. The emergence of powerful computer software and models supporting SE activities led to an evolution towards model-based systems engineering (MBSE). Becoming more and more prevalent, MBSE shows a lot of potential, such as improvements in communications among stakeholders, in capturing project knowledge or in the ability to manage system complexity. As a result, the MBSE approach delivers an unambiguous model of the system. This model is a central and unique artifact of the system development process [INC-2015, p. 189; Fri-2014, p. 20].

With increasing application of MBSE in product development, the need for an integration of Product Lifecycle Management (PLM) functions in the digital system model appears and is discussed in literature [Kir-2017]. Further approaches to close the gap between the model as result of a function-oriented systems thinking approach and conventional product development tools for design and simulation exist [Moe-2015]. The relevance of this aspect for terminal planning is shown by *Lange and Kastner*, who criticize the lack of dynamic process simulation possibilities during the static layout planning phase of intermodal terminals [Lan-2019].

In a nutshell, the problem addressed by this thesis is that a plausible and realistic architecture for the NGT CARGO logistics terminal has to be developed. As stated previously, the need for the integration of digital models in development processes exists and solutions to establish ties with conventional software emerge. An application of the MBSE approach towards such a logistics facility was not found in literature.

1.3 Research Question and Objectives

It is obvious that a development of an intermodal logistics terminal is a complex engineering problem. MBSE addresses such problems and thus, is an interesting approach to support and to improve the system architecture development process of an intermodal logistics terminal.

This thesis is embedded in the NGT CARGO logistics project conducted by DLR. For reasons of scope, the targeted MBSE approach was intended to focus on the architecture facilitating intralogistics freight handling within the NGT CARGO logistics terminal.

The research question addressed within this thesis is therefore formulated as:

How can MBSE successfully be applied to guide the system architecture development of an intermodal freight terminal in context of the NGT CARGO logistics project?

To give an answer to the research question in regard to the selected scope, this thesis directs at two research objectives as displayed in Figure 1-5.

The first research objective aims at the demonstration, how an architecture for the NGT CARGO logistics terminal can look like supported by a guiding MBSE approach. It implies an investigation on intralogistics transshipment processes and the development of an

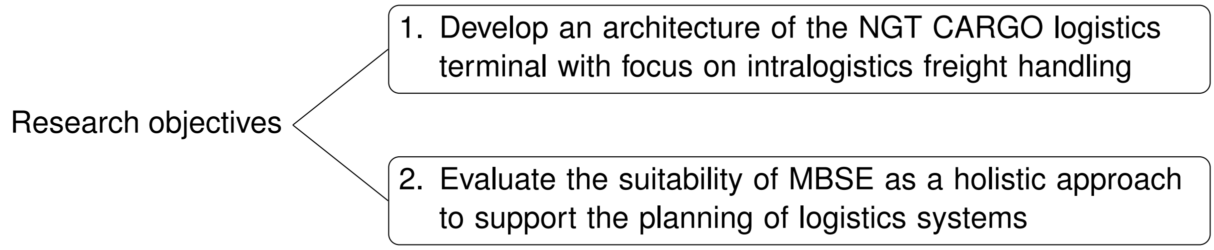


Figure 1-5: Definition of research objectives

architecture capable of facilitating the required processes. The logistics terminal's architecture shall be modeled focusing on its intralogistics core and following an appropriate MBSE methodology. The outcome is a verified terminal system architecture demonstrating a possible technical implementation of the NGT CARGO logistics terminal.

The second research objective aims at the demonstration, that the application of MBSE was successful and how such an approach can contribute to future logistics plannings. It is addressed by validating the outcome of the architecture specification process in regard to its applicability from a logistics perspective. The suitability of the MBSE approach is to be elaborated considering experienced benefits or difficulties. Further, a comparison of the conventional standard approach for logistics terminal planning and the MBSE approach under investigation shall be executed on a basic level. The outcome is a validated terminal system architecture and a discussion of a potential integration of MBSE in the logistics planning.

1.4 Research Approach and Thesis Outline

The guiding research methodology applied within this thesis is based on the Design Research Methodology (DRM) which was firstly introduced by *Blessing and Chakrabarti* in 1992 [Ble-2009]. Due to its generic character the DRM is a research design guideline which can be applied in various disciplines and covers various types of design research. It aims to support the planning and designing of research by providing process steps and thus, to facilitate more valid and useful results. It is important to state that the DRM is not a sequential process model to follow stage by stage rather than an iterative framework of research methods and guidelines [Ble-2009, pp. vii, 10–11].

As macro structure for research projects, the DRM proposes a framework with the four stages *research clarification* (RC), *descriptive study I* (DS-I), *prescriptive study* (PS), and *descriptive study II* (DS-II) [Ble-2009, p. 15]. The tailoring of the four stages and their basic means and outcomes in context of the following work are illustrated in Figure 1-6.

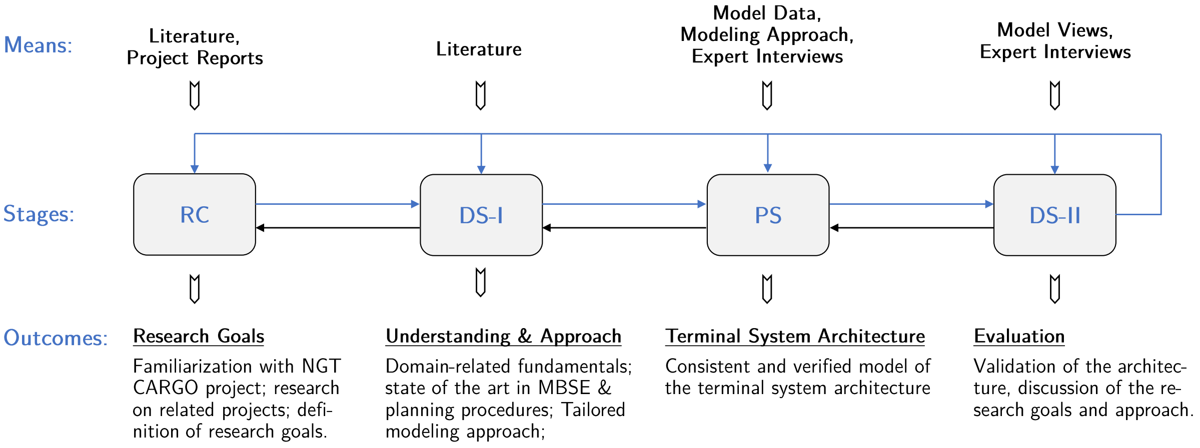


Figure 1-6: Application of the DRM framework in context of this thesis (own illustration based on Blessing and Chakrabarti [Ble-2009, p. 15])

The basic intention of the *RC* stage is to identify the overall research plan, containing among others the research question, corresponding objectives, and the approach. The literature is reviewed to indicate lack of research and to position the own research project [Ble-2009, pp. 29–31].

The second stage of the DRM, namely the *DS-I*, targets a deeper understanding of the existing situation. Familiarization with fundamental concepts and existing work related to the research objectives is key and complement respectively update the deliverables from RC. Implications of the literature review for the upcoming development of support in next stage are gathered [Ble-2009, pp. 31–33].

The *PS* is the stage where a solution to the described problem is created. This refers to the development of support for improving design. Focus is laid on coherent documentation. Concluding the PS, a verification is carried out as first evaluation of correctness of the support [Ble-2009, pp. 33–35].

The fourth stage of the DRM framework is the *DS-II*. It focuses on the evaluation of the developed support as outcome of PS and its ability to realize the research goals stated in

RC. It serves as recapitulation of the conducted research and aims at a lessons learned conclusion of the project [Ble-2009, pp. 35–38].

For the application of the DRM on different research projects, *Blessing and Chakrabarti* identify seven types of research designs [Ble-2009, p. 60]. According to this categorization the following thesis equals a research design type 3 (*development of support*[2]). Specifically, the terminal system architecture model is regarded as the actual support which has to be developed. In type 3 research designs, RC and DS-I are review-based and enable an adequate understanding of the problem so that a comprehensive PS is facilitated. The support developed within the PS is subject to an initial evaluation in DS-II [Ble-2009, p. 61].

The *structure of this thesis* follows the stages of the DRM. Figure 1-7 gives an overview of the involved chapters, their key contribution, and a classification to the DRM framework.

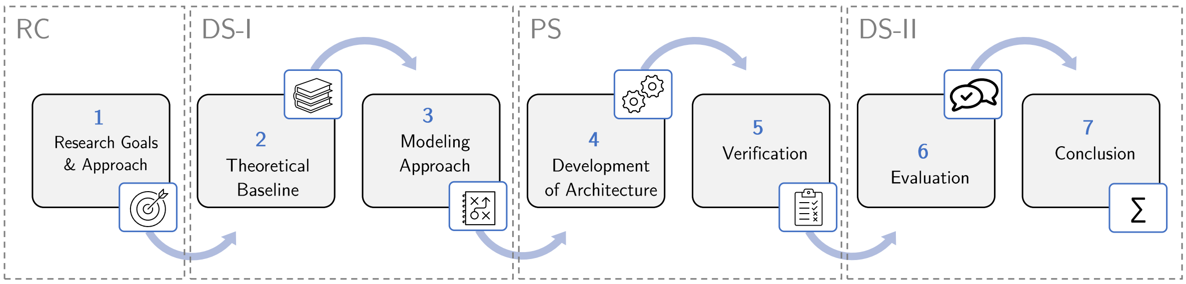


Figure 1-7: Outline of the thesis, with basic classification of chapters to the stages of DRM (own illustration)

Chapter 1 covers the RC stage. In this thesis, the RC was driven by the familiarization with the NGT CARGO project to clearly understand the basic environment and problem description of this thesis. Further, the initial phase was mainly based on research on intermodal terminals and MBSE, to sharpen the research question and to derive the research goals. The involved means were domain-related literature and project reports of the NGT CARGO project. The first chapter is concluded by the research approach, guiding the structure of this thesis.

The research question defined in th RC stage requires understanding of three fundamental topics: MBSE, intermodal terminals, and intralogistics freight handling. Facilitating this

[2] The term 'support' is used by *Blessing and Chakrabarti* as collective term for possible means or measures, such as strategies, methodologies, procedures, techniques, or software tools [Ble-2009, p. 4].

understanding, **Chapter 2** addresses the key task of the subsequent DS-I and gives an overview about fundamental concepts and reviewed literature. In addition, detailed introduction to previous work on the logistics terminal by DLR in relation to the position of this thesis is given, and the related state of the art in MBSE and logistics planning is elaborated. The implications of literature guided the definition of modeling tasks and a tailored modeling approach, which are described in **Chapter 3**. This chapter gathers the enhanced understanding and directions from the literature research and delivers a plan to approach the development of support.

As mentioned before, the actual support developed within the comprehensive PS stage refers to the specification of the terminal system architecture. Based on the fundamental theory, modeling data and the modeling approach from the DS-I stage, the development of the architecture is presented by **Chapter 4**. Driven by the MBSE approach, the created system model serves as coherent documentation providing views of the specified architecture. Concluding the PS stage, the derived terminal system architecture and the model itself are verified applying the means of an expert interview and SysML inherent verification capabilities. This verification is described in **Chapter 5**.

As this thesis is classified as type 3 research design, an initial DS-II stage was realized. Thus, the evaluation in **Chapter 6** focuses on the indication of the applicability of the specified architecture. Therefore, the validation of the previously verified terminal architecture was executed based on the model views and expert interviews. In the evaluation context, the extent to which the research objectives set in the RC phase could be achieved with this thesis is discussed. A subsequent evaluation of the research approach serves as lessons learned conclusion from the conducted research. Finally, **Chapter 7** summarizes the results obtained. The thesis concludes with an outlook and suggestions for further work.

2 Theoretical Baseline

Basic knowledge about the main areas of subject is required to understand the issues addressed by this thesis. The following chapter's intention is to introduce theory underlying the concerned topics and to give an overview over related literature.

This thesis targets readers with various background knowledge. Thus, the first section aims at establishing a common understanding of fundamental concepts regarding MBSE, intermodal terminals, and intralogistics. Furthermore, the results of the previous work on the NGT CARGO logistics terminal by DLR are regarded as fundamental (Section 2.1). Based on this elementary baseline, Section 2.2 puts the spot on the applied core concepts of MBSE and the state of the art of intralogistics planning processes. A short summary in section 2.3 wraps up the theoretical guidance and draws conclusions from the state of the art.

2.1 Related Fundamentals

Firstly, Subsection 2.1.1 focuses on MBSE and system models, while the following parts elaborate on intermodal terminals (Subsection 2.1.2) respectively intralogistics freight handling (Subsection 2.1.3). Concluding the section of related fundamentals, Subsection 2.1.4 briefly wraps up the results of the previous work on the NGT CARGO logistics terminal by DLR.

2.1.1 Fundamentals of MBSE and System Models

To impart the meaning of MBSE works best by firstly talking about its origin in SE and the evolution from a document-centered SE approach to a new, digital practice of engineering based on system models.

The roots of **SE** trace back to the 1930s. It appeared firstly in the defense industry, and later became an accepted practice in the aerospace industry as well. Engineers had to develop solutions to technologically challenging and mission-critical problems while facing a rapid increase in complexity. A successful way of cope with the challenges was found by establishing the systems concept in engineering [Hug-1998; INC-2015, pp. 12–13]. More recently, other industries started to recognize the value added by systems thinking and the application of the holistic and interdisciplinary SE approach [INC-2014]. Today, the SE practice is standardized [IEE-1220; ISO-15288] and broadly documented [SEB-2020], ensuring a unified reference in terms of terminology, measures, and tools [Ram-2012].

The concept of SE bases on two ideas - system thinking and process models (Figure 2-1).

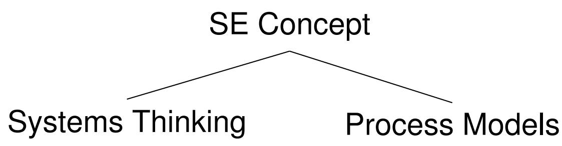


Figure 2-1: Decomposition of the SE concept in systems thinking and process models [Hab-2019]

Systems Thinking

In systems thinking context, a **system** is defined as a *'combination of interacting elements organized to achieve one or more stated purpose'* [ISO-15288]. This definition refers to a human-made, target orientated configuration of subordinated system elements being a kind of hardware, software, data, human, process, or procedure. According to the *International Organization for Standardization*, a system is further characterized by [ISO-15288]:

- a defined system boundary, marking out the system itself from its environment
- a defined hierarchical or other type of relationship among the system elements
- the possibility to regard an entity at any hierarchical level within the system as a system by its own

Applying this systems concept on a specific engineering problem means that the subject to engineer is regarded as one system with specified border, containing related elements and an interacting with a surrounding environment. The system being under consideration is called **System-of-interest (SOI)** [ISO-15288]. It is important to note that various SOI may exist within one engineering project, depending on the individual role of stakeholders. Thus, a SOI can be defined more generally as *'the system of interest to an observer'* [SEB-2020; Ber-1968]. The entity of the SOI's environment and the interacting relationships are denoted as system context [SEB-2020; Flo-1993].

If any element being part of a SOI is seen as a system by its own, rather than being treated as a bottom-level blackbox element, this element becomes a subsystem of the SOI with own elements, relations, and environment. In the other direction if systems are merged within a top-level system the more comprehensive system is termed as suprasystem. The multiple application of the concept of sub- and suprasystems leads to an ever-increasing detail respectively abstraction level of the system, resulting in a system hierarchy [Hab-2019, pp. 6–8].

The ability to consider an engineering problem in a holistic manner, defining boundaries according to a specific SOI to which the systems concept is applied is the key idea of **systems thinking** [SEB-2020]. Figure 2-2 visualize the systems concept principle.

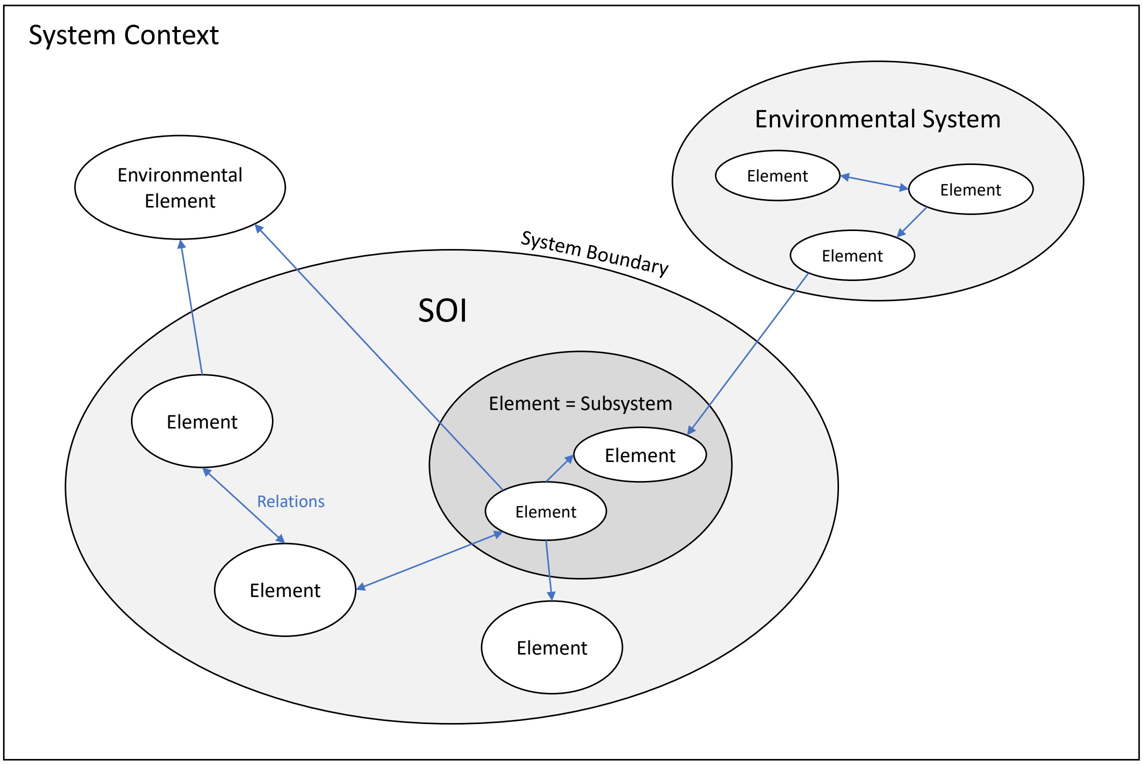


Figure 2-2: Visualization of basic systems thinking terminology (own illustration based on Haberfellner et al. *and* International Organization for Standardization *[Hab-2019, pp. 5, 7; ISO-15288, p. 11])*

SE Process Models

A variety of general process models, so called life cycle models, and their corresponding activities are discussed in the literature, such as e.g. the Waterfall [Roy-1970], Vee [For-1991] or SIMILAR [Bah-1998]. An overview over existing process models is given by *INCOSE* or *Haberfellner et al.* [INC-2015, pp. 32–37; Hab-2019, pp. 56–78]. Usually, process models specifying engineering activities are kept abstract to decouple the process itself from subject-specific issues . However, domain-specific process models exist, e.g. for development of software, electric, electronic, or mechanical systems [Eig-2017].

In SE multidisciplinarity plays a major role for developing balanced system solutions. Thus, a **SE process model** covers all technical aspects of system development as well as the management aspects. It focuses on understanding the multi-domain stakeholder's needs and specifying the resulting requirements in an early stage of the system development process. The motivation behind this is to detect and avoid errors or misunderstandings as fast and reliable as possible [Fri-2014, p. 4].

Based on *Hall* and *Daenzer*, the SE process model 'Hall/BWI' is described by *Haberfellner et al.* exemplary for SE process models [Hal-1962; Dae-1977]. Compared to previously stated, general process models, it seeks for a more comprehensive approach by being capable in creating new solution concepts as well as considering situation-dependent adoption of existing good ideas. It consists out of the following four basic principles [Hab-2019, pp. 27–55]:

- *Top-down procedure*: This principle strongly relates to the system hierarchy thinking and implies that SE shall proceed step-by-step from a general level to more detailed levels.
- *Creation of variants*: SE seeks for the best system solution, which can be ensured by thinking in alternatives instead of settling for the first available solution. If a top-down procedure is followed, thinking in variants should not involve significant additional effort.
- *Phased approach*: The whole system development process is subdivided into time-structured and staged phases. This procedure addresses especially the management aspect of SE. The entity of phases serves as macro-logic providing structure and ensuring on time joint decision making. General process models can serve as phase guidance within the SE process model.
- *Problem-solving cycle (PSC)*: The PSC designates a guiding procedure to cope problems regardless of type or phase as lowest level of iteration (micro cycle). In this context, the term 'problem' refers to a discrepancy in actual and desired state. To overcome this discrepancy the PSC suggests three sequential stages, namely 'search for objectives', 'search for solutions' and 'decision'.

These four components are regarded as being representative for SE process model principles. The application of SE is driven by the combination of these four components to create a holistic systems approach. Although the SE process model 'Hall/BWI' is regarded to be a plan-driven method it supports the application of agile development to certain extend [Hab-2019, pp. 79–82].

Limits of SE and Transition to MBSE

With rising size or scope of the initial problem to engineer, the application of SE concepts induces several problems. As size increases, so does the amount of information to be considered. Growth in scope leads to a diversification of expertise needed. In a document-based approach this information is spread across many documents e.g. spreadsheets, text or drawings. Inconsistencies and lack of re-use of information are to be expected [SEB-2020]. Based on *Holt and Perry*, *Moeser et al.* identify rising complexity, lack of understanding and communications as the greatest challenges of SE [Hol-2013; Moe-2015].

Another problem of document-based system development is more related to organizational change in engineering problems. With rising functionality of products, the amount of domains involved increases. Neither the quantity of information nor the variety of domain-specific expertise of complex systems can be captured by single persons anymore, as it was done by the chief-engineer in charge for a long time. High fluctuation of staff brings up the challenge to transfer existing knowledge from the individual employee into a sustainable and accessible form within the company [Mug-2021].

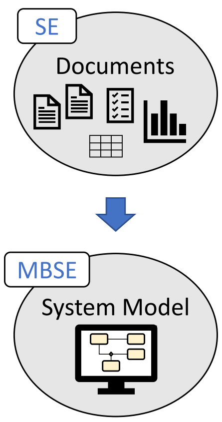


Figure 2-3: Evolution from SE to MBSE (own illustration)

The evolution from the traditional, document-based SE towards MBSE came along with increasing computer technology (Figure 2-3). The first models for mechanical engineering e.g. for computer-aided design (CAD) emerged in the 1980s. However, it took until the 1990s when it was realized that the usage of consistent system models was a great facilitator to cope the rising challenges in SE business, too. MBSE as term was firstly documented by *Wymore* in 1993, who published a purely mathematical formalism for a model-based approach towards systems [Fri-2014, p. 16; Wym-1993]. More than 20 years later, the

International Council on Systems Engineering (INCOSE) defines **MBSE** as *'the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases'* [INC-2015, p. 189].

Even if MBSE is still in an early maturation stage, the recent INCOSE vision expects MBSE to become a standard practice for successful SE activities by 2025 [INC-2014, p. 38].

System Models and MBSE

In his introduction to model theory, *Stachowiak* proposes that a **model** can be regarded as an image of the original object, pragmatically reduced to the essential attributes in the modeled context. When working with models, these three basic properties of models have to be kept in mind [Sta-1973, pp. 131–133]:

- *Property of imaging*: A model always relates to an original entity, which is natural or human-made in origin or even a model itself.
- *Property of reduction*: In general, a model exhibits only a part of the original entity's attributes. The certain extend of reduction is set by the model creator or user according to his*her perception of relevance.
- *Property of pragmatism*: A model does not claim to be universally valid or applicable. It always refers to a defined purpose (being reason) requested by a specified human or artificial user for a certain frame of time.

These basic properties of modeling theory apply to system models in MBSE context. Thus, a **system model** initially represents an abstract proxy of the real system and is the key artifact of MBSE. The more the MBSE process moves ahead, the more concrete the system model gets depending on purpose, stakeholders, and frame of time. The model can be regarded as incrementally, but constantly evolving representation of the system throughout the system's life cycle [Mad-2018].

Apart from the general definition of models, a system model is further specified as an interconnected set of the four core elements *specification*, *design*, *analysis*, and *verification*. These core elements determine the system's requirements, structure, behavior and parametrics [Fri-2014, p. 17]. A representative visualization of the system model's core elements and their cross-cutting relationships is given by Figure 2-4.

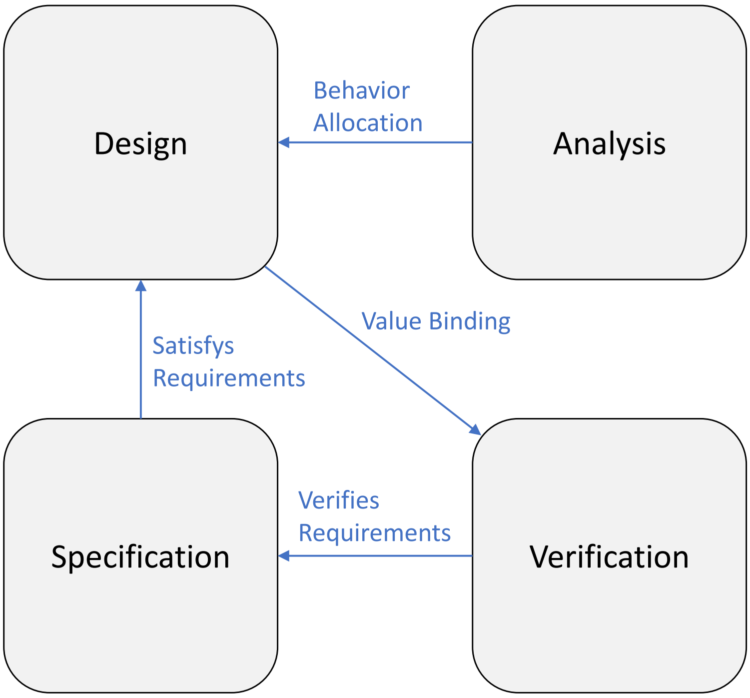


Figure 2-4: Visualization of a representative system model (own illustration based on Friedenthal et al. *and* Steiner *[Fri-2014, p. 17; Ste-2015])*

The key paradigm of system models in MBSE context is the separation of data respectively information and their visual representation. The information is unique and stored in the model repository (single source of truth). The visual representation is realized by views. In a view the information is displayed graphically in various ways. This allows the model user to consider only the information of interest from an individually desired perspective. Especially in multi-domain projects views are useful to reduce visual complexity and to facilitate clear communication. As a background baseline the model repository manages the information and maintains consistency among the different views [Fri-2014, p. 17; Bra-2020, p. 34].

A significant added value to the engineering process by using system models is to enable the design of a system which is kept consistent in satisfying its requirements and meeting its overall objectives. Furthermore, working with a holistic and coherent system model induces improvements in transparency and traceability of system design, in addressing various stakeholders by creating discipline-specific views and in keeping in line with cost and time-to-market goals [Fri-2014, pp. 17–20; Mad-2018, p. 186]. Once the model is defined and reached the desired level of detail, dynamic system simulations can be executed based on the provided system model [Par-2010].

Developing System Models

Although there are many ways leading to the goal of a comprehensive but manageable system model, they all share a common core concept. Based on *Friedenthal et al.*, *Dickopf et al.* state that creating a system model in MBSE context basically requires three components: a *modeling methodology*, a *modeling language*, and a *modeling tool* [Dic-2017; Fri-2014]. This trinity concept is known as the **three pillars of MBSE**. As visualized in Figure 2-5 MBSE needs every pillar for a successful system model development. The modeling tool is used to perform a set of tasks expressed in the modeling language, while being instructed by the modeling methodology [Del-2014, p. 4].

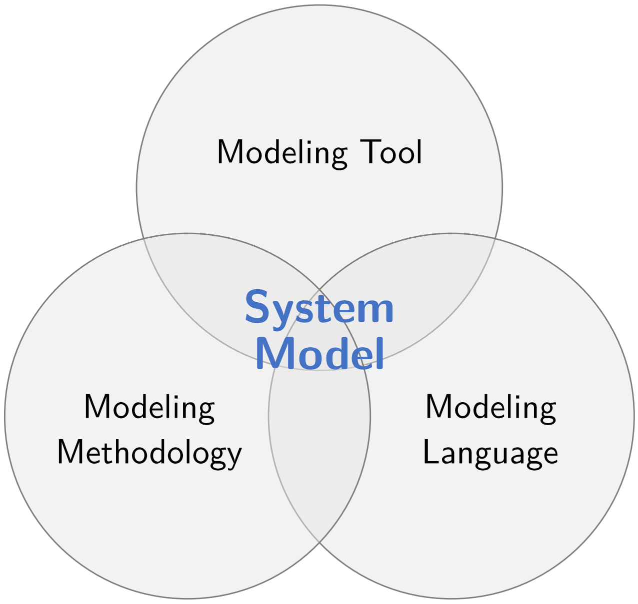


Figure 2-5: Three pillars of MBSE as key elements of system model development (own illustration)

A **modeling language** is a semiformal medium for communication in development processes [Del-2014, p. 5]. A modeling language's core is a specification containing basic rules for its syntax and its semantics. The syntax defines the structure of the language's words or symbols as set of characters. It can be understood as rules how the language is visualized based on textual or graphical elements. Contrary, the semantics defines the meaning and interpretation of the symbols supplied by syntax. It is important to notice, that in case of domain-specific languages semantics may differ according to system context [INC-2015, p. 186; Bra-2020, pp. 23–24].

Modeling languages are used to describe systems in a textual or graphical way. Usually, they are standardized to provide an unambiguous definition of semantics and syntax. Since graphical views of models are easier to comprehend than text-based code, most system models today are based on graphical modeling languages [Alt-2012, p. 28].

As a holistic guidance a **modeling methodology** instructs how to use artifacts provided by the modeling language. It represents a road map describing how to proceed in order to obtain a purpose-oriented system model [Del-2014]. Triggered by different conceptions of meaning, certain misunderstandings related to denotation of *methodology* occur. Partly based on *Estefan* the following definitions are applied within this thesis [Est-2008]:

- A **process** is a logical arrangement of tasks (= WHAT) to fulfill a defined purpose or goal without defining how the tasks are done.
- A **method** specifies HOW each task is done by providing suitable techniques.
- A **methodology**, as it is part of the three pillars of MBSE concept, is determined to be the collection of related processes and methods applied to support the discipline of SE in a model-based context.

The third pillar of MBSE is represented by the **modeling tool** which is used to create and work on the model. The system model is expressed using a modeling language, but the modeling tool visualizes the output. According to the type of supported modeling language(s), the tool must be capable of generating code or graphical diagrams. By making use of the modeling tool, the system model data is created and managed in the repository, and views are generated. Thus, the tool can be regarded as the interface between the systems engineers (creators of the model), stakeholders (the users of the model) and the system model itself [Alt-2012, pp. 66–67].

However, modeling tools are much more than just diagramming tools for illustrating syntax. In contrast to simple diagramming tools the consistency of views and data is ensured by constantly checking the system model underlying the visualization. This helps the systems engineer to create a well-formed system model which conforms to the modeling language's rules. In case of changes in elements within one diagram all views are updated with the new information due to the commonly shared set of data [INC-2015, p. 188; Del-2014, p. 8].

2.1.2 Fundamentals of Intermodal Freight Terminals

The NGT CARGO is an intermodal transportation concept including the NGT CARGO logistics terminal which this thesis is about. Before approaching the theoretical background of intermodal terminal facilities, the term 'intermodality' has to be clarified.

Intermodal Freight Transportation

Intermodality is a specialization of Multimodality. **Multimodal freight transportation** is a form of transportation where two or more modes of transportation are involved in the transportation chain of freight (Figure 2-6). The basic idea of combining different modes of transportation within one shipping activity is to take advantage of the individual mode's beneficial properties. On long-haul transportation loads are consolidated efficiently by rail or ship while the local pick-up and delivery operations are executed using flexible trucks. Fast air transport gets involved especially in time-critical transportation processes on long distances. As a result, the overall efficiency of the whole distribution process is improved significantly [Bek-2007].

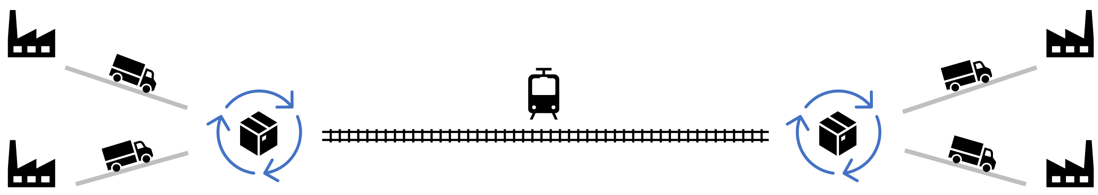


Figure 2-6: Example of a multimodal transportation chain (own illustration)

The efficiency of multimodal transportation can be further increased if the process of changing transportation modality is simplified. Firstly demonstrated by Malcom McLean in 1956, transshipping containers instead of individual loads reduces the excessive handling effort and thus, cuts the transshipment costs and time [Mon-2017b, p. 3].

This led to the development of **intermodal freight transportation** which is defined as *'(...) movement of goods in one and the same loading unit or road vehicle, which uses successively two or more modes of transport without handling the goods themselves in changing modes'* [Uni-2001].

Typical load units used for intermodal transport are containers, swap bodies or semi-trailers (Figure 2-7). A container is a box made out of steel and is standardized in terms of measures and fastening devices. Being similar to containers, swap bodies are metal superstructures which are additionally equipped with folding support-legs. Containers and swap bodies can be transported on most lorries or railway wagons with flatbed chassis as well as on vessels. In contrast, semi-trailer denominates a wheel equipped trailer, which is to be attached to a semi-trailer tractor. The elevated position of swap bodies and semi-trailers enables trucks to detach and mount the load unit without external loading facility [Wox-1998, p. 2].

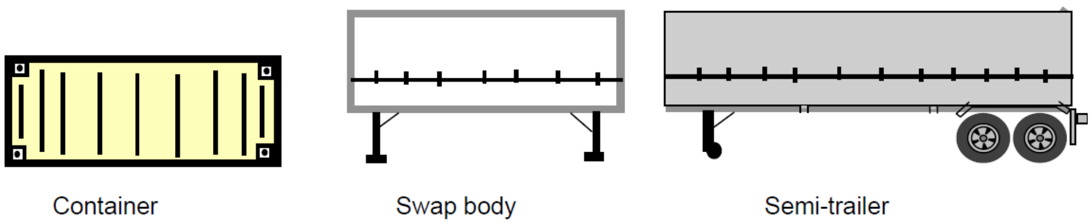


Figure 2-7: Load units used in intermodal transportation [Wox-1998, p. 3]

Intermodal Transshipment

The place where the intermodal interchange of these load units takes place is the **intermodal freight terminal**. By definition, the primary function of intermodal terminals is to facilitate the process of interchanging the load units and to provide an interface for all modes of transport involved. To tranship load units different technology is used depending on the mode of transport as well as on the type of load unit. Examples are reach-stackers, gantry cranes, or roll-on/roll-off ramps [Lam-2006, pp. 22, 30].

Usually, intermodal terminals concentrate on this transshipping process rather than serving an origin-/destination-based market. Although, a storage facility is not necessary in theory, in practice a storage place or short-time buffer usually is maintained [Mon-2017a, p. 9]. Terminal operations may also contain sorting and consolidation of load units. In case of international freight traffic some terminals involve customs and security services and thus, load unit scanning facilities are provided [Bek-2007].

According to *Lampe*, intermodal terminals can be structured in four basic subsections. Figure 2-8 shows the layout of a generic intermodal terminal for transshipment from road to rail and vice versa [Lam-2006, p. 52].

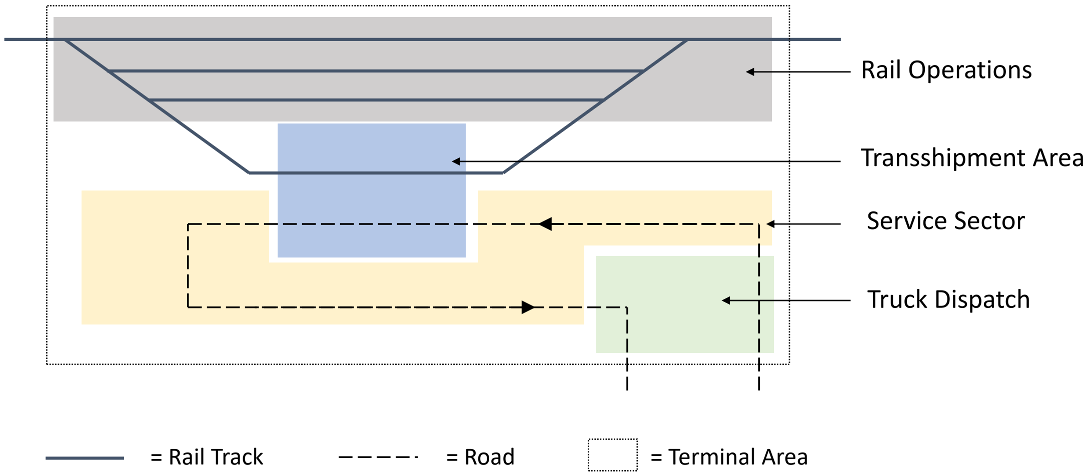


Figure 2-8: Basic layout and modules of an intermodal container terminal with rail/road transshipment (own illustration based on Lampe *[Lam-2006, p. 52])*

Core of the facility is the *area of transshipment*. Within this module, the physical transshipment process is executed and the rail vehicles are unloaded respectively loaded. In case of direct transshipment, the load unit is immediately transshipped from rail wagons to trucks (or vice versa). In case of indirect transshipment, short-time storage for load units is provided.

In the *service sector* further short-time and long-term storage area is supplied. This is especially used for swap bodies or semi-trailers, which can be directly and independently picked up by trucks without unnecessarily blocking the transshipment area.

The *rail operations* module denotes the rail track network provided by the terminal for operating, handling, shunting, or parking locomotives and rail wagons. Due to the system-inherent complexity of train handling shunting yards occupy large areas. The rail operations area represents the connection from the transshipment area to the rail track network and contains the entry and exit gate for trains.

The fourth subsection is the *truck dispatch*. As its counterpart on the rail side, the truck dispatch provides the area for handling and parking/buffering of trucks. The entry and exit gate for trucks and thus, the connection to the road network is located in this section. Arriving and leaving vehicles are registered and the administration is done [Lam-2006, pp. 52–53].

Using *MegaHub Lehrte* as an example, Figure 2-9 illustrates a cross section of the core transshipment area of a state-of-the-art rail/road terminal. *MegaHub Lehrte* was inaugurated in 2020 by DB Netz AG. Its core elements are the crane system and the fully automated sorting system. The transshipment facility provides six train handling tracks, short-time buffer areas, and driving respectively loading areas for road vehicles [Deu-2017].

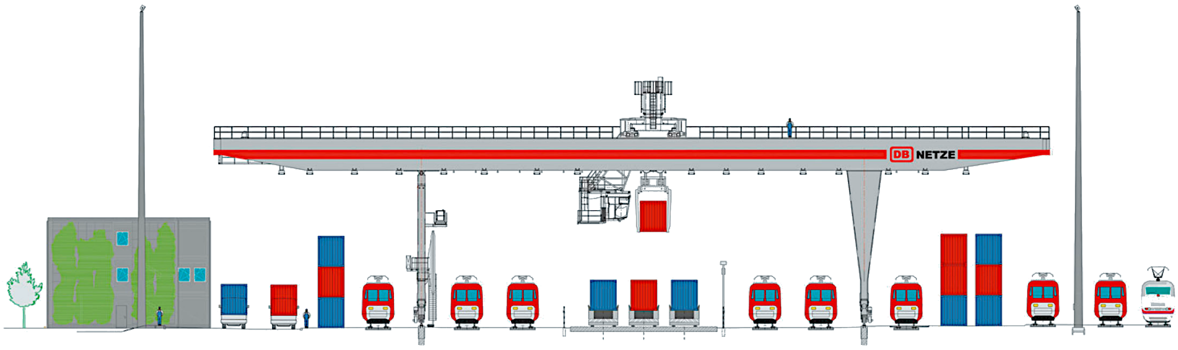


Figure 2-9: Cross section of the MegaHub Lehrte as a state-of-the-art example of intermodal container terminals [Deu-2017]

2.1.3 Fundamentals of Intralogistics Freight Handling[1]

There are four core operational functions in logistics: *transportation*, *transshipment*, *storage* and *picking* of goods. These functions interact in logistics networks to ensure an efficient supply of customers with the required quantities of goods needed at the right time in the right place. Logistics nodes are major components of these logistics networks. Their main purpose is to gather inbound procurement flows and to create outbound distribution flows. In between, goods run through intralogistics processes [Gud-2010, pp. 3, 21].

Intralogistics Processes

According to *Arnold*, **intralogistics** in general can be described as a bundle of tasks dealing with in-house material flows, information flows, and handling of goods. These tasks include management, controlling, implementation and optimization. Due to this rather generic specification intralogistics refers to many logistics facilities with a huge variety of processes [Arn-2006, p. 1]. In consequence of the thesis' scope, in the following the focus is laid on transshipment facilities rather than intralogistics in production systems.

[1] Various literature about logistics was reviewed in German language. Since no standardized translation guide was found, translations for logistics terms in this thesis were taken from *Hompel and Heidenblut* whenever available [Hom-2011].

From an intralogistics perspective a logistics node can be classified as distribution center that basically receives, temporarily stores and forwards goods [Wis-2009, p. 1]. *Alicke et al.* introduce the Distribution Center Reference Model (DCRM) in order to provide a benchmark baseline for all types of distribution centers [Ali-2006]. A more detailed elaboration on the DCRM is given by *Wisser* [Wis-2009].

On process level, the DCRM delivers a generic breakdown of the overall intralogistics process. Six basic **intralogistics processes** of distribution centers are identified (Figure 2-10).

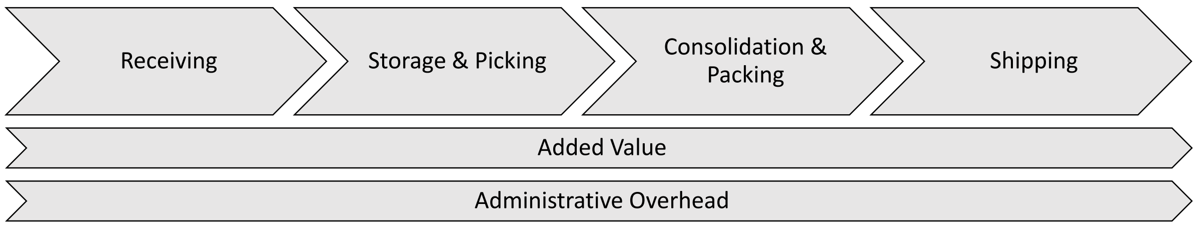


Figure 2-10: Basic intralogistics processes in distribution centers (own illustration based on Alicke et al. *and* Wisser *[Ali-2006; Wis-2009, pp. 11–16])*

Inbound goods firstly follow the *receiving* process. Incoming goods are unloaded from the transportation vehicle, identified, and allocated to an inbound buffer area. As soon as the following inbound control checks are completed and the receipt is approved, the freight is cleared for further processing.

Subsequently, the goods are transported to the next section and enter the *storage and picking* process. Usually, this represents the key process of distribution centers. For storage, goods are put into the storage location, stored, and retrieved when requested [Ali-2006; Wis-2009, pp. 11–13]. Picking is strongly tied to storage activities and refers to the composition of specific quantities of articles from a provided total assortment based on orders [VDI-3590]. The storage and picking process is terminated by providing the goods as required for further transshipment.

In a next step, the *consolidation and packing* process follows up. The merging of goods to one unit with same target destination is done either on the way to the packing station or within a specified sorting system. Subsequently, the goods are identified and prepared for transportation. Depending on the goods, a multi-staged process execution is possible. As for the previous processes, the consolidation and packing process ends with the provision of the goods for the next step.

The last sequential generic process in distribution centers is the *shipping*. It mainly aims at the handover of goods to the mean of transportation. Before the intralogistics process ends with the loading of goods, they arrive at the outbound buffer area, are identified, and sorted according to the corresponding vehicle.

The *added value* process summarizes possible additional services provided by the distribution center. Examples for such services are price marking, labeling, or assembly services. Due to the variety of additional services offered by logistics facilities, no generic sequence allocation within the material flow activities is given. However, to integrate added value processes in the generic material flow an interface to the area of add-value has to be provided at the right place.

Apart from sequential material flow processes, a superior process appears in distribution centers. The *administrative overhead* gathers activities which are not directly related to the execution of the material flow, such as registration, supervising, directing, or steering processes. Example tasks are dispatch and allocation of transportation vehicles, staff scheduling, or facility management [Ali-2006; Wis-2009, pp. 13–16].

This is a generic approach to processes in logistics nodes. *Wisser* emphasizes that not every distribution center must exhibit all six processes. For example, in case of a *cross-docking* procedure, the goods are transshipped within a logistics node without storage or picking activities [Wis-2009, p. 12].

Load Units in Intralogistics

In contrast to conventional intermodal transportation where containers, swap bodies, and semi-trailers are used as top-level load units (see Subsection 2.1.2), lower leveled load units are of interest in intralogistics transshipment processes.

According to DIN 30781, a **load unit** consists out of the good itself (liquid, gas, bulk, or piece goods) and a load carrier bundling the good into one piece. On a lowest level a load unit in logistics is denoted as package and refers to any transportable packaging holding the goods together [DIN-30781]. To increase handling performance several packages can be gathered and secured on other load carriers. Together, packages and the load carrier are regarded as next leveled load unit. These Load units are usually named according to the load carrier used [Mus-2013, pp. 127–128].

A huge variety of **load carriers** for intralogistics transshipment exists [Hom-2018, p. 13; Gud-2010, p. 419]. Load carriers and especially their interface for intralogistics transportation systems are often standardized to enable a usage along the transportation chain. Figure 2-11 visualizes two exemplary common load carriers, namely the pallet and the box pallet.

Figure 2-11: Intralogistics load carriers; Euro pallet (left side) and box pallet (right side) [Eur-2021]

One of the most important load carriers is the *pallet* (Figure 2-11, left). Different variations of pallets exist, such as the European pool pallet (Euro pallet, 1200 mm x 800 mm) or the UK-pallet (industry pallet, 1200 mm x 1000 mm). In Europe, a pallet pool exist which enables an exchange of standardized pallets under consideration of certain criteria.

Another common type of load carrier in intralogistics is the *box pallet* (Figure 2-11, right). Basically, the box pallet is a pallet with an attached top frame made out of lattice bars. The frame contains an opening for withdrawal of the contained packages or goods [Mus-2013, pp. 128–130].

Intralogistics Systems Components

To facilitate the transshipment of the load units within a logistics terminal, physical intralogistics systems are required. Plenty of physical intralogistics components are described in literature. In the following, basic characteristics of *conveying systems*, *sorting systems*, *storage systems*, and *order picking systems* are briefly presented.

Conveying Systems

In general, a system moving goods or people is denoted as transportation system. In case of in-house or on-site processes, the movement is specified as conveying activity. Thus, **conveying systems** in intralogistics are subsystems facilitating the physical material flow within a logistics facility. They move goods or load units among the different functional sections from one or more sources to one or more sinks. Depending on the particular system design, the movement is horizontal, vertical, or both. Apart from the conveying function itself, conveying systems can also be used for distributing, sorting, buffering, or short-term storing of load units. The logical combination of different conveying systems results in an intralogistics material flow system [Hom-2018, p. 125]. A classification of conveying systems is made based on their dynamic behavior. On top-level two conveying system categories are differentiated, namely *continuous*[2] and *discontinuous* conveyors [Aßm-2019, p. 2].

Continuous conveyors are fixed facilities, which are characterized by constantly conveying goods in particular intervals without having the load handling attachment to return to the starting point for the next item to transport [Hom-2011, p. 296]. Roller conveyors and chain conveyors are two prevalent examples for continuous conveyors (Figure 2-12). Con-

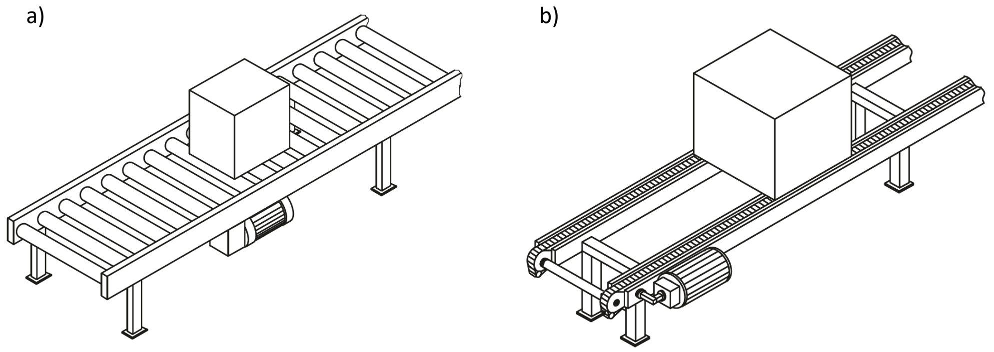


Figure 2-12: Continuous conveying systems; roller conveyor (a) and chain conveyor (b) [Hom-2018, pp. 136, 147]

tinuous conveyors are particularly beneficial in energy efficient and economical conveying of relatively large quantities of goods in a short period of time. The simple structure and predefined routes result in a deterministic material handling and thus, in advantages in operational safety, operations, and suitability for automation. By design, continuous conveyors

[2] *Aßmann* remarks that a constant conveying may refer to steady and quasi-steady material flow. However, he defines that 'continuous conveyor' denotes both, steady and quasi-steady conveyors [Aßm-2019, p. 2]. For the purpose of this thesis, this definition is adopted.

offer certain buffer functionality due to topology-related long conveying paths or particular accumulation segments (Figure 2-16, d). Problems to consider when using continuous conveyors are low flexibility in routing or pick-up interfaces, limited capacity expansion possibilities or creation of obstruction for humans or other moving devices. Furthermore, in case of connecting different sections of the logistics facility, continuous conveyors may cause issues in fire protection or cool chain [Hom-2018, pp. 103, 133–134].

Discontinuous conveyors are movable devices, which are characterized by transporting load units in discretely sequential respectively simultaneous motions (work cycles) using a dedicated load handling attachment [Hom-2011, pp. 320–321]. Figure 2-13 shows different

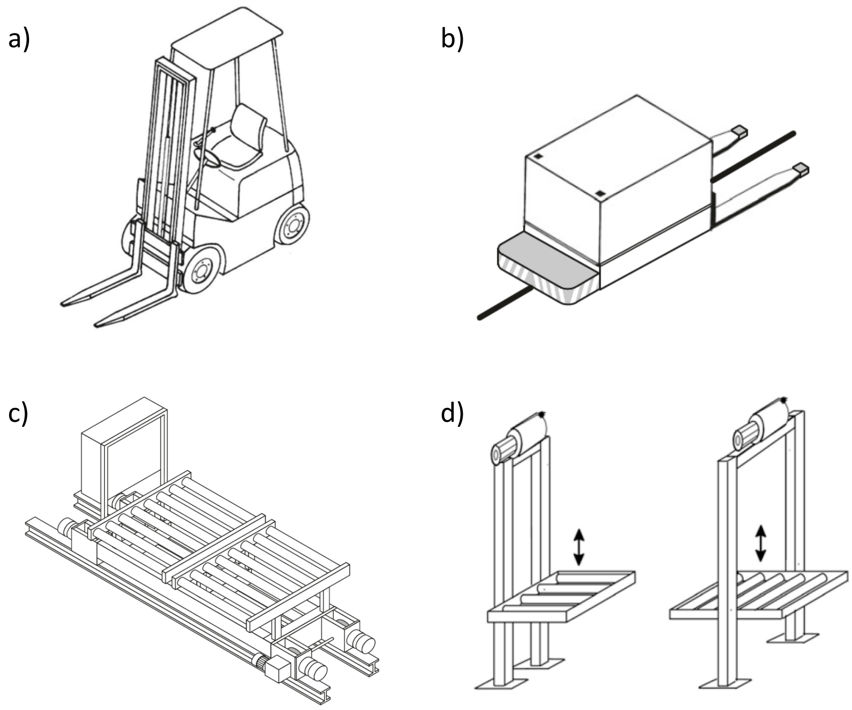


Figure 2-13: Discontinuous conveying systems; forklift truck (a), automated guided vehicle (b), transfer carriage (c) and elevator (d) [Hom-2018, pp. 178, 207, 211; Aßm-2019, p. 23]

examples for discontinuous conveying systems, such as a forklift truck and an automated guided vehicle (AGV) for multidirectional or loading activities, a transfer carriage for lateral movement and elevators for vertical conveying. The main advantage of this type of conveying system is its adaptability and flexibility. By choosing the right system and load handling attachment nearly every conveying task is realizable. The overall performance of the system is adjustable by increasing or decreasing the number of individual devices. Consequently, capacity enhancements or creations of additional pick-up locations are possible with rather low effort. In general, sources and sinks of the material flow are easily served without additional transshipment devices. However, increased effort in allocation and guiding systems is required. Further problems are the hazard of collisions and high investment costs [Hom-2018, pp. 161–163].

Sorting Systems

Whenever various load units with different sources respectively sinks are transshipped within one intralogistics system, a **sorting system** is necessary. A sorting system denotes several coupled or parallel conveying facilities technically realizing the distribution of the goods according to the individual intralogistics destination. Complex sorting systems with various multi-staged components exist for example in courier, express, and parcel industry (CEP), production systems, or airport baggage handling. These complex, multi-staged facilities mainly focus on packages or small goods. However, sorting of pallets as load units can be realized by the use of distributing and consolidating interfaces among different conveyor systems within a transshipment facility [VDI-3619; Beu-2019].

Free-floating discontinuous conveyors, such as forklift trucks or AGV (Figure 2-13 a,b) are capable of sorting the particular load by simply heading to corresponding sources or sinks [Hom-2018, p. 161]. Fixed systems such as roller or chain conveyors (Figure 2-12) need additional elements for feeding and the transfer of load units. Depending on the type of conveyor different interface elements exist [Hom-2018, pp. 261–266]. Three examples of transfer interfaces used for roller conveyors are illustrated by Figure 2-14. The chain transfer unit (a) or the turn table (b) are used for direction changes or transfer between continuous conveyors, while the infeed/pick-up unit (c) is used for transfer between continuous and discontinuous conveying systems.

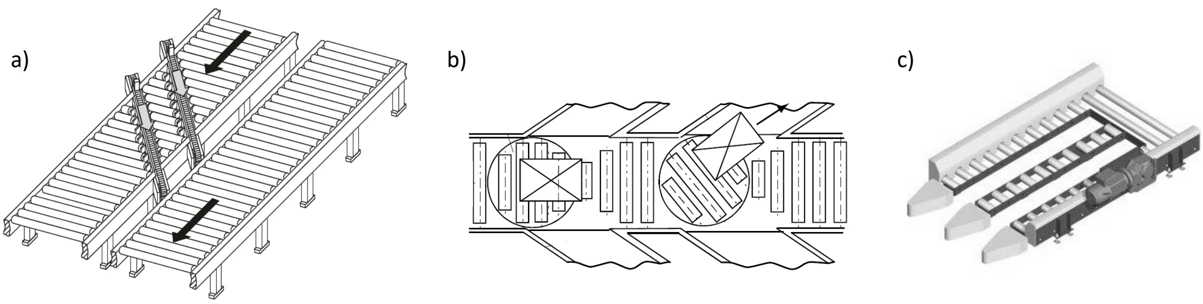


Figure 2-14: Transfer interfaces for continuous conveyors; chain transfer unit (a), turn table (b), and infeed/pick-up unit (c) [Hom-2018, pp. 139, 147; ATS-2021]

Storage Systems

The overall task of a warehouse is to hold goods until further processing is requested. The related intralogistics processes are putting load units into storage, holding load units in storage bins, and retrieve load units from the storage. Core of a warehouse is the **storage system**. Unless it is not operated manually, a storage system is served by a storage and retrieval device. The interface section between the warehouse itself and the intralogistics transportation system is denoted as warehouse interface area [Hom-2018, pp. 51–56]. A principle sketch of a warehouse is illustrated by Figure 2-15.

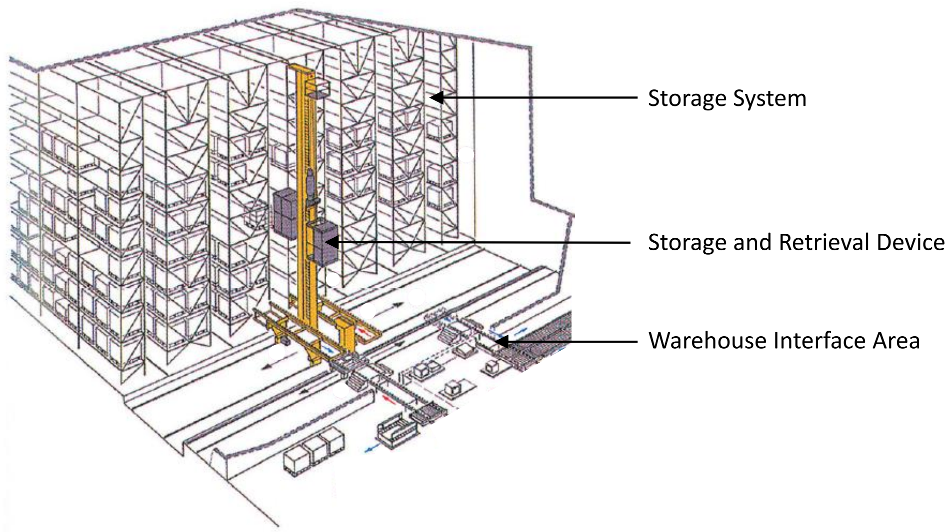


Figure 2-15: Warehouse with storage system, storage and retrieval device, and warehouse interface area [Arn-2019a, p. 197]

Storage systems are classified based on various criteria. Basically, goods can be stored on the *ground*, in *static rack-based* or *dynamic rack-based* systems, and on *conveyors* [Hom-2018, p. 58]. Figure 2-16 shows a representative example for each category.

Ground storage (Figure 2-16, a) is the simplest type of storage system. The load units are placed on the floor and stacked on each other. In the spatial arrangement, block storage (without aisles) and line storage (with aisles) is differentiated. Advantages are high space utilization, high flexibility towards local conditions, and low investment costs. Yet, not every load unit suits storage without racks for static reasons and basically only last-in first-out (LIFO) strategies are possible when using ground block storage [Hom-2018, pp. 60–64]

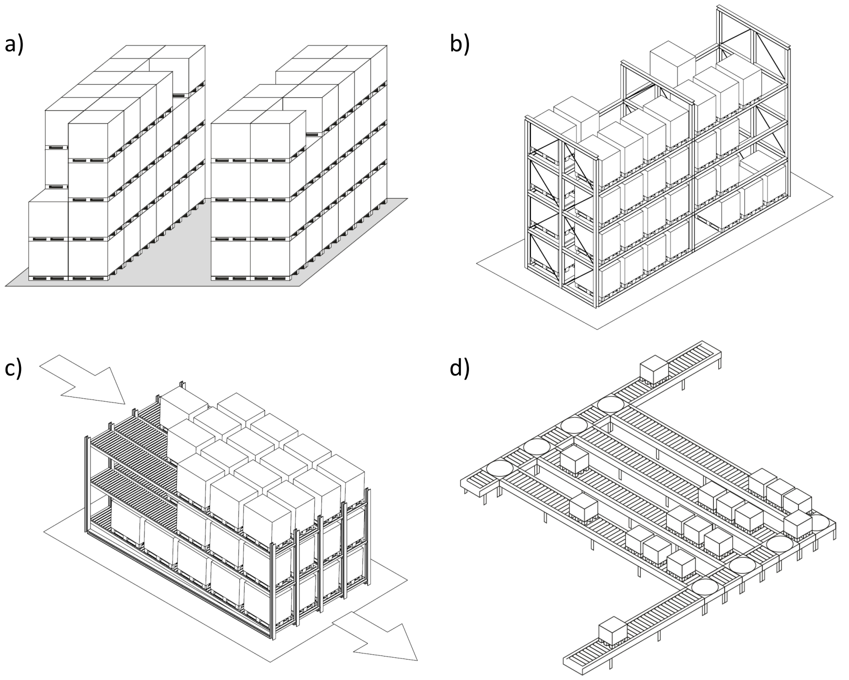


Figure 2-16: Storage systems; ground line storage (a), pallet rack (b), flow rack (c) and accumulation roller conveyor (d) [Hom-2018, pp. 63, 93; Sch-2019b, pp. 80, 87]

The most common representative of **static rack-based storage systems** used for goods on load carriers is the pallet rack (Figure 2-16, b). Dependent on the particular design, single bay respectively multi bay and single-deep respectively double-deep systems are differentiated. High-bay racks are pallet racks with a height greater than 12 m and can reach up to 55 m in height. Pallet racks are easy to automate and support any storage strategy since every load unit can be accessed (except for double deep storage). Disadvantages exist in terms of fixed storage bin size, lower space utilization compared to ground block storage, and higher investment costs [Hom-2018, pp. 66–68; Sch-2019b, p. 79].

In **dynamic rack-based storage systems**, either the load unit itself or the whole rack is moving. The flow rack is an example of static racks with moving load unit (Figure 2-16, c). The storage bins are designed as sloped or driven corridors, equipped with either roller or chain conveyors. Load units are stored from one side and move to the other side while storing keeps going on. The first load unit arriving on the other side is the one to retrieve first. Thus, flow racks have to be operated in exclusively article-only lanes and enforce a first-in first-out (FIFO) storage strategy. Despite this, they are easy to automate due to the deterministic behavior and support high throughput rates [Sch-2019b, pp. 86–87].

The final category of storage system is the **storage on conveyors**. As already touched on in the conveyor section, this type of storage mainly refers to short-time storage and buffering activities. In representation for various types of conveyors capable in buffering load units, Figure 2-16 (d) shows an accumulation roller conveyor. Key aspect of accumulation roller conveyors is that the short-term storage is seamlessly integrated into the conveying system and thus, no additional storage and retrieval device is needed. Compared to sloped flow racks, less dynamic pressure affects the load units [Hom-2018, pp. 93–94].

Strongly tied to storage systems are **storage and retrieval systems**. Basically, these devices are conveyor systems as introduced before. Since not every storage system is compatible to every storage and retrieval device, a complementary pair has to be taken into account when designing storage systems. To give an idea of storage and retrieval systems, Figure 2-17 illustrates three different technologies for this type of device.

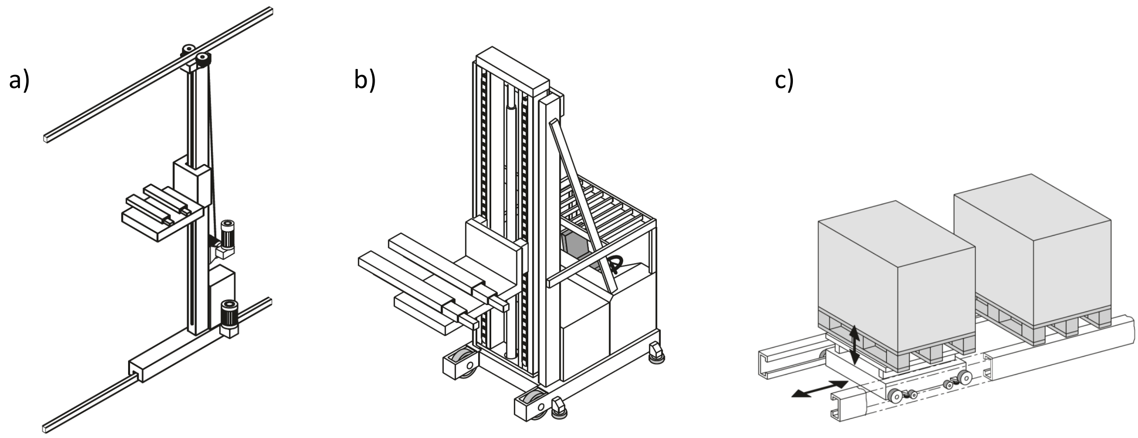


Figure 2-17: Storage and retrieval systems; stacker crane (a), tri-lateral high rack stacker (b) and satellite shuttle (c) [Hom-2018, pp. 183, 198, 225; Aßm-2019, p. 26]

Picking Systems

Picking systems facilitate the order picking process which was introduced in the process section before (Figure 2-10). These systems are tied to a warehouse or equipped with a designated picking storage system. Due to the complexity of the process, picking activities are often executed manually. On top level, picking systems are differentiated by the movement behavior within the process (Person-to-Goods/Goods-to-Person) [Arn-2019a, pp. 229–232]. Although picking is claimed to be a key service of logistics centers, it is out of modeling scope within this thesis and thus, not covered here in detail. Reference to related literature for picking systems is given [Hom-2019; Arn-2019a, p. 239; Gud-2010, pp. 659–759].

2.1.4 Previous Work on the NGT CARGO Logistics Terminal

To give an overview about the project-related context of this thesis, Figure 2-18 illustrates the position of *previous work*, *contribution of this thesis* and the *following work* within the schematic Vee process model as part of the NGT CARGO logistics terminal project by DLR.

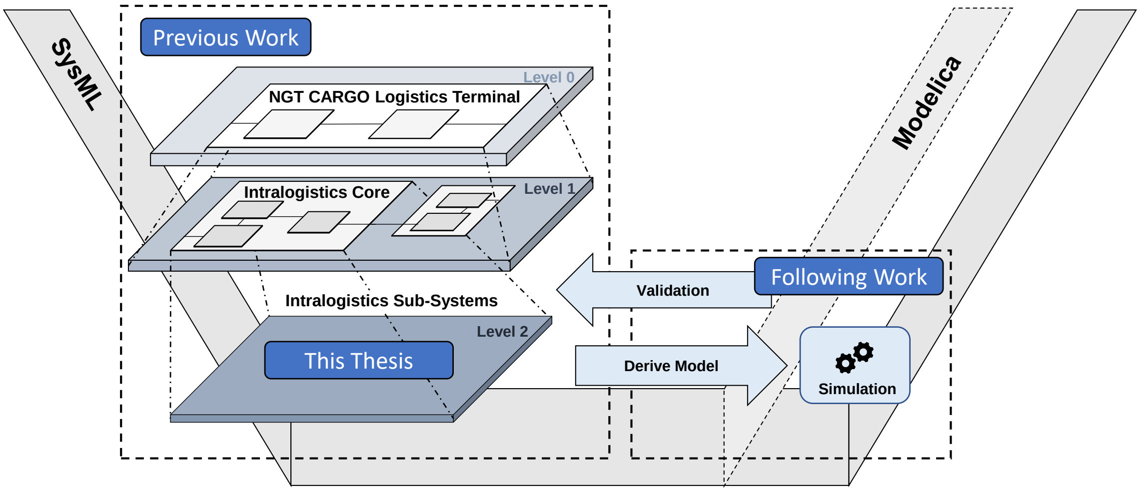


Figure 2-18: Project context of this thesis within the development of the NGT CARGO logistics terminal (own illustration based on DLR)

For the development of the NGT CARGO logistics terminal an MBSE approach was followed. A system model was initialized using the modeling language SysML and the modeling tool Cameo Systems Modeler. Within previous work by DLR, a system analysis of the total logistics terminal was executed. Based on the results of the system analysis, the contribution of this thesis is the design of the system architecture of its intralogistics core. For subsequent validation of the developed terminal, a simulation of the model using Modelica is planned within the following work.

The previous work on the MBSE approach towards **the terminal's system analysis** was published in detail by *Ehret et al.* and *Malzacher et al.* [Ehr-2020; Mal-2020]. In the following the results of this system analysis are briefly presented based on these publications. The systems analysis was split into four analysis processes: *system context analysis*, *stakeholder analysis*, *requirement analysis*, and *use case analysis*.

Within the **system context analysis**, the initial top-level objective framework of the logistics terminal was specified. The terminal was denoted as major part to be integrated into possible logistics chains for LDHV goods. Due to high requirements in regard to transport times, reliability and costs, the terminal was identified as bottleneck within the logistics chain. As basic conclusion of the system context analysis it was found that the logistics terminal mainly interacts with three different categories of external systems: *logistics management systems*, *railway operation systems*, and several *means of transport*. Further, the need for an internal management system to organize the internal workflows and the communication with external systems was stated.

Subsequently, the **stakeholder analysis** revealed further insights about the stakeholders interacting with the terminal. Main stakeholders are the *users and operators*, such as forwarders, carriers (road, rail, others), logistics service providers, haulers, and terminal and infrastructure operators (road, rail, energy, communication). Other persons or organizations affected by the terminal were classified as *public* (customs, railway authorities) or *developing* (research and concept design, construction, subsystem developer). Figure 2-19 shows the system context with identified stakeholders.

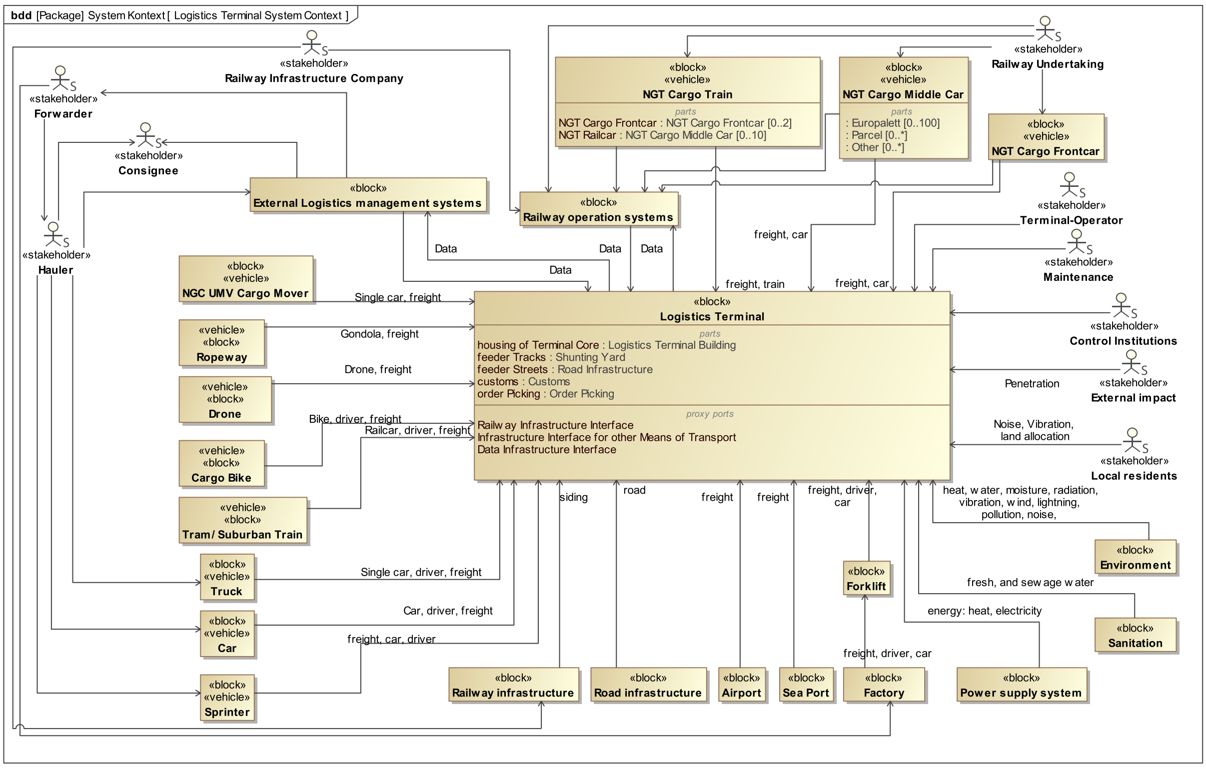


Figure 2-19: System context of the logistics terminal with stakeholders and affected systems [Ehr-2020]

The **requirement analysis** of the terminal was conducted using literature reviews and stakeholder interviews. For the whole terminal 50 top-level requirements were defined and split into the categories *functional*, *usability*, *reliability*, *performance*, *physical* and *business* requirements. A key outcome was that the requirements strongly relate to the type of good which is transshipped in the terminal. Different **groups of goods** were defined, as illustrated by Figure 2-20. A more detailed specification of the groups of goods is given in Appendix A. The goods were found to have a major impact on handling and intralogistics processes, mainly driven by differences in regard to possible load carrier, standardization potential, or required freight conditions. These differences resulted in the specification of functional requirements regarding specifically the *treatment of goods*. A single terminal facility to transship every good was denoted as hard to achieve.

Figure 2-20: Relevant LDHV groups of goods for the NGT CARGO logistics terminal (own illustration)

As a last step of the systems analysis, the **use case analysis** resulted in the definition and partly specification of 23 use cases (UC). Based on the identified UC, a top-level system process was developed (Figure 2-21). The system process specifies the general workflows for the overall NGT CARGO logistics terminal operations. It shows various activities, which can be clustered in *road vehicle operations* (left side), *NGT CARGO operations* (right side), *terminal management* (top and center), and the *terminal core operations* (bottom center) [Ehr-2020; Mal-2020]. This top-level process concludes the description of the previous work as project-related baseline of this thesis.

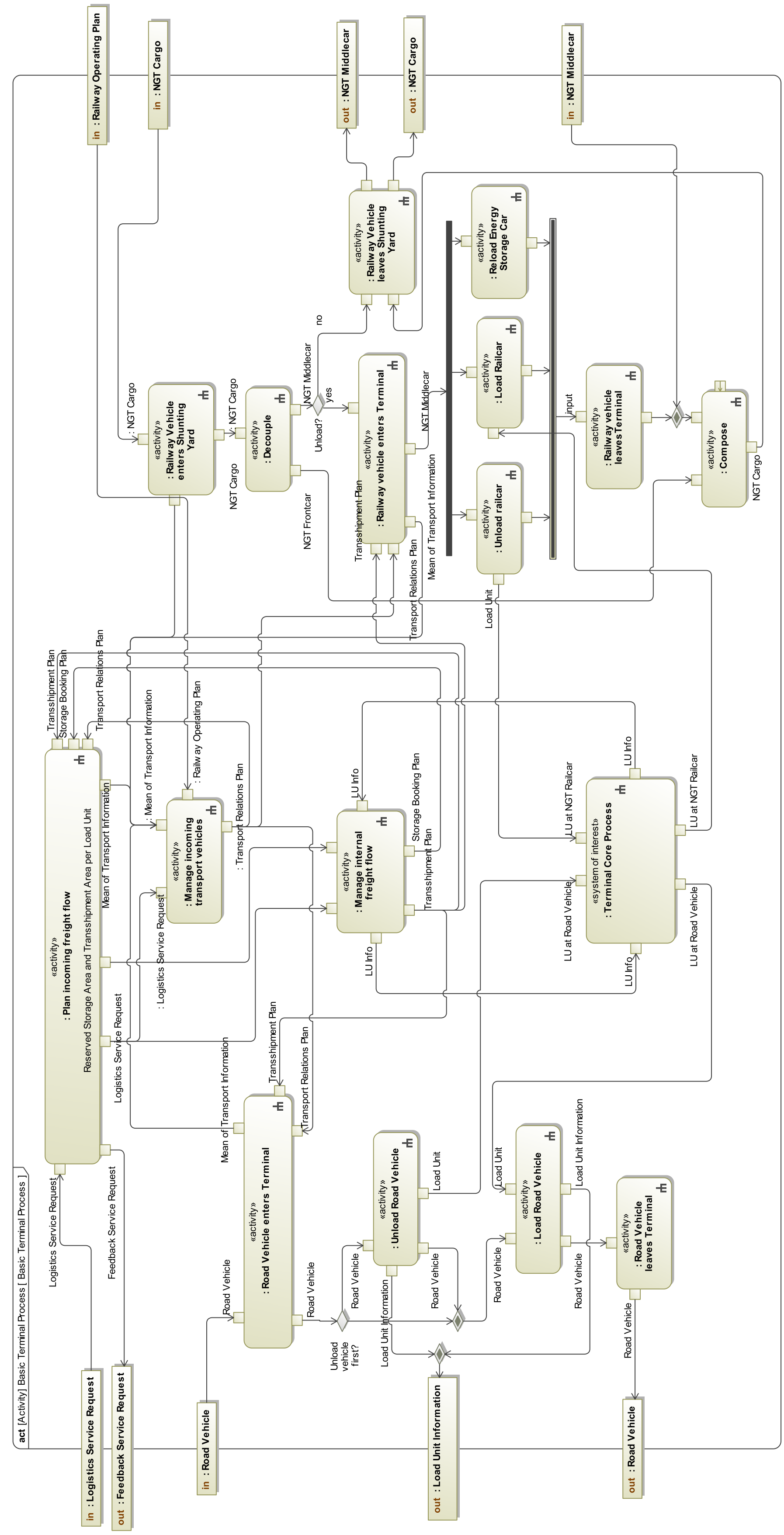


Figure 2-21: Top-level system process of the NGT CARGO logistics terminal [Ehr-2020]

2.2 State of the Art

After fundamental theory was clarified in the previous section, the following part refers to additional literature review in order to address the research objectives stated in Section 1.3. To develop the architecture of the intralogistics core system using MBSE, a system model is required. Therefore, Subsection 2.2.1 introduces state-of-the-art implementations of MBSE core concepts for system model development. Subsection 2.2.2 elaborate the current planning process of intralogistics systems as subject of comparison to the MBSE approach.

2.2.1 Implementation of System Models and Selection Criteria

As described in Subsection 2.1.1, the system model is the primary artifact of MBSE relying on three core concepts: modeling language, methodology and tool. To realize the system model development, for each pillar a corresponding implementation had to be selected. In the following, state-of-the-art implementations are briefly presented for each concept before selection criteria are given.

Modeling Language

One of the first modeling languages to be standardized was the Integration Definition for Functional Modeling (IDEF0) in 1993. Originally designated for software functional modeling, its main incentive was to provide a generic and precise modeling language which is independent of computer-aided software engineering [Nat-1993]. Shortly it was followed by the Object-Process Methodology (OPM) which is a conceptual approach to automation systems providing a bimodal language (textual as Object-Process Language and graphical as Object-Process Diagrams) as well as a methodology [ISO-19450]. In 1997, the Object Management Group (OMG) adopted the Unified Modeling Language (UML) which is a language concept driven by object-oriented programming. Over the years, the UML evolved up to UML 2.5 and is a worldwide accepted standard for object-oriented software engineering [Obj-2017]. A few years later, in 2002 the UML was the starting point for the 'UML for Systems Engineering' initiative. This project was renamed and finally adopted by OMG in 2007 as Systems Modeling Language (SysML) [Obj-2019; Mug-2020a]. Today, SysML is a broadly accepted, general-purpose dialect (profile) of the UML, specifically developed for (MB)SE application and standardized by *International Organization for Standardization*

[ISO-19514]. Another recent development is the ARChitecture Analysis and Design Integrated Approach (ARCADIA) which was originally developed by Thales. It is a modeling methodology bringing up its own same named language concept and focusing on design of system architectures. ARCADIA is a domain-specific modeling language and is mainly based on UML/SysML. Its intention is to facilitate acceptance of MBSE by stakeholders who are not familiar with generic modeling languages [Roq-2016; Voi-2018].

This list is not conclusive. Other modeling languages exist, offering alternatives for efficient approaches in specific context. However, SysML is establishing as the generic-purpose standard modeling language for MBSE [Ram-2012; Wei-2016a; Del-2014, p. 10].

Modeling Methodology

A variety of different modeling methodologies exists. In a survey-based benchmark, *Estefan* gives an overview about existing modeling methodologies applied in an MBSE context [Est-2008]. In a more recent benchmark from 2016, *Weilkiens et al.* lists further methodologies [Wei-2016a]. Exemplary, identified modeling methodologies are:

- ARCADIA (see modeling languages) [Voi-2018]
- IBM Telelogic Harmony-SE [Hof-2011]
- IBM Rational Unified Process (RUP) for SE [Mur-2001]
- Object-Oriented Systems Engineering Method (OOSEM) [Fri-2014, p. 417; Lyk-2000]
- OPM (see modeling languages) [Dor-2002; ISO-19450]
- Requirements-Functional-Logical-Physical approach (RFLP) [Kle-2013]
- Systems Modeling Toolbox (SYSMOD) [Wei-2014; Wei-2016b]

Further, methodologies focusing on the development of functional architectures only exist, such as the Functions-Based SE method (FBSE) [INC-2015, p. 190] or the Functional Architectures for Systems method (FAS) [Lam-2014; Wei-2016c].

However, these methodologies are not to be regarded as fixed, rigid procedures demanding for comprehensive adherence. *Ramos et al.* denote the methodological principles as rather informal, yet expect them to mature and to establish as norms or best practices in the future [Ram-2012]. Additionally, it is important to note that these methodologies are generic description of procedures and thus, not every step may be of interest to the system the methodology is applied to. Some of them address specific modeling purposes (FAS, FBSE), while others deliver a full body (end-to-end) approach towards system models (OOSEM) [INC-2015, p. 190]. Therefore, tailoring or combinations of methodologies can be taken into account [Del-2014, p. 7]. This is confirmed by the results of a survey among MBSE users performed by *Cloutier and Bone* in 2010, where 37% of the participants stated that company-specific methodologies were applied [Clo-2010, p. 127].

Modeling Tool

Modeling tools are provided by both, commercial vendors and nonprofit organizations. They differ in many factors, such as compatibility with modeling languages, costs, capability, available add-ons, or adaptability [Del-2014, p. 8]. A survey conducted in 2010 by *Cloutier and Bone* revealed that a variety of tools is used in industry (in this case especially tools for SysML application). According to the results of this survey, the three most used tools were Rhapsody by IBM Rational, NoMagic MagicDraw, and SparxSystems Enterprise Architect, which are all vended modeling tools [Clo-2010, p. 117]. Further commercial tools are listed by *Delligatti* such as the NoMagic Cameo Systems Modeler or Artisan Studio by Atego. However, with growing community in MBSE, open-source modeling tools emerge. Using an Eclipse application as workbench Modelio, Papyrus, and Capella are representatives of free modeling tools [Del-2014, p. 8; Roq-2016]. Further open-source basic diagramming tools and flowchart makers such as diagrams.net exist. Although they do not provide full model support and thus, are not suitable for professional system models, they can be used for fundamental MBSE visualizations and support familiarization with modeling languages and methodologies [Bra-2020].

Again, this list is not conclusive and other products which were not named previously provide excellent solutions as well. A comprehensive and up-to-date record of existing tools is given by *Weilkiens* [Wei-2020b].

Selection of Concepts

Few literature was found on guiding the choice of which modeling language to use. However, learning to apply the language in a confident way is key to successful modeling and requires adequate literature support [Bra-2020]. Further, it must be assured that the modeling language is appropriate for the type and domain of system to design [Del-2014, p. 5].

For the initial selection of a tool, various approaches are described in literature [Jon-2011;

Ros-2018]. Possible evaluation criteria are modeling language support, usability, simulation capabilities, generation of documents, adaptability, expandability and licensing [Alt-2012, pp. 68–70]. A huge variety of further criteria exist, mainly depending on the requirements of application [Fri-2014, p. 539; Ros-2018]. A comprehensive comparison using a modeling example is recommended, but involves a lot of effort. Further, reviews and evaluation benchmarks can be consulted [Jon-2011; Piv-2021].

Weilkiens et al. criticize that the focus in the selection process of MBSE pillars is laid on tools rather than on methodologies. Due to an identified lack of existing selection support for methodologies, they introduce an evaluation framework for MBSE methodologies. *Weilkiens et al.* state that a practical MBSE methodology has to be focused on product development, provide a comprehensive documentation, and must have reached a certain degree of maturation. A variety of further possible criteria is given by *Weilkiens et al.*, structured according to support, usability, efficiency, practicality, and essentials. Concluding it is said that selecting a methodology is subjective and one-to-one comparison is not always useful due to differences in coverage of processes according to ISO 15288 [Wei-2016a; ISO-15288].

Conclusive Remarks on Selecting MBSE Pillars

By design, tools are always related to one or more modeling languages. Yet, as already touched on, further relations among the three pillars exist. In case of OPM, the modeling methodology comes up with a corresponding modeling language [Dor-2002]. An even more integrated MBSE approach is the explicitly associated trinity concept with a corresponding language, methodology and tool provided by ARCADIA/Capella [Roq-2016]. On the other hand, regarding methodologies, some modeling languages like UML/SysML are completely agnostic to the chosen concept and can be combined with any methodology [Est-2008].

It is important to emphasize that particular implementations of the three concepts can do both, complement each other or even lead to contradictions and thus, making a modeling impossible. Therefore, facilitating the comply of the system model's purpose without discrepancies among the pillars is key to successful modeling [Moe-2015]. The selection of MBSE pillars always refers to the selection of a complementing set of implementations. However, the selection of pillar components appropriate to the particular MBSE project is difficult and requires a lot of expertise [Wei-2016a].

2.2.2 Planning Processes for Logistics Facilities

This section is intended to give an overview about the state of the art in logistics planning and corresponding potential for improvement. Before presenting state-of-the-art planning approaches for logistics facilities, general aspects of current logistics facility planning are briefly introduced.

Logistics Facility Planning

Planning in general denotes a systematic business process including the specification of objectives and all necessary future activities and corresponding sequences to achieve the desired results [Hom-2018, p. 339]. Companies usually provide a top-level strategic corporate planning, which is the long-term baseline for all planning activities [VDI-3637, p. 2].

Logistics facility planning is a major concept within this framework. It covers planning activities for *production* (in case of producing facilities), *material and information flow*, *building structure and energy supply*, and *communication and office systems* [VDI-3637, p. 3]. These planning activities contain the selection, arrangement, and configuration of suitable logistics systems from a wide range of options for each planning area, so that the specified performance requirements are met at optimum cost, taking into account all boundary conditions [Gud-2010, p. 67].

Planning of logistics facilities is initiated for many reasons. Examples for planning objectives are an enhancement in capacity, efficiency, or reliability, as well as cost reductions in terms of staff or tied capital or simply entering new business markets [Hom-2018, pp. 339, 340]. Consequently, logistics facility planning projects are classified in four different categories [VDI-5200, p. 4]:

- *Development Planning*: Also called 'greenfield site' planning, where a brand-new logistics facility is built with respect to terrain-specific restrictions rather than having existing building structures to be considered.
- *Replanning*: Denotes an expansion or optimization project based on an existing logistics facility with various boundary conditions, such as existing building structures or continuous on-site operations.
- *Clearance/Demolition*: Refers to the closure of a logistics facility, which shall be dismantled to initialize a possible reuse of the site.

- *Revitalization*: Also called 'greyfield site' planning, where abandoned industrial facilities or sites are considered to be prepared for re-utilization.

The content of logistics facility planning projects can be further distinguished based on concerned *planning domains* and *levels*.

The fundamental planning scope is described by the targeted **planning domain**. Every planning project starts with the domain *objectives*, where the corporate strategy is transformed into logistics objectives, constraints, and assumptions. The three main planning domains are the search and selection of a *location*, an integration into *external logistics*, and the *factory and production logistics* in narrow sense, denoting the planning of the intralogistics system itself and its interfaces to external infrastructure.

Furthermore, five **planning levels** are differentiated. The chosen planning level refers to the hierarchy in the company's logistics business. The level of planning content can be set to various hierarchy layers of interest, such as a single *work center* or *production segment*, up to a *building*, a *logistics site* or even a whole *production network*.

The selected planning domain and level are fundamental to the logistics planning approach, as they determine the overall project scope. Different planning domains or levels require a different focus or level of detail within the planning process [VDI-5200, p. 6; Dom-2018, pp. 55–56].

Logistics Facility Planning Processes

To enable a time-efficient and solution-effective planning of logistics facilities a systematic approach according to proven methods is key for the planning process. Thus, a planning process in general is organized in several phases containing particular working steps. These phases and their working steps are passed through until the specified objectives are achieved and all requirements are met [Gud-2010, p. 69]. The general consecutive description of these phases must not be understood as straight waterfall process to follow. Instead, they describe an iterative process. In case of new findings or changes in the ongoing planning, a return to previous phases including rework loops may be required [Hom-2018, p. 348].

As a blueprint process framework, four **general planning phases** can be defined [Fot-2020]. These general planning phases are fundamental to the state-of-the-art approaches and are therefore shortly elaborated in the following. The logical sequence of the four phases is illustrated by Figure 2-22.

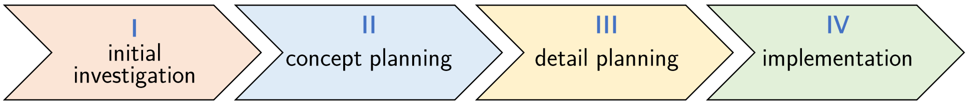


Figure 2-22: Generic phases of planning processes for logistics facilities [Fot-2020]

The first stage of a logistics planning process is the **initial investigation phase**. Firstly, a project plan, the planning tasks, and the overall objectives are defined. These deliver the basic input for further planning project management and define a common baseline of expectations against which the planning result is measured. Subsequently, an evaluation of the current state is performed. This includes a procurement, preparation, and analysis of data, such as actual quantities, processes, resources as well as layouts, buildings and real-estate of interest [VDI-5200]. By evaluating the collected data, the feasibility of achieving the previously defined objectives is examined, vulnerabilities are revealed, and a list of necessary measures is generated. Based on the increased fundamental understanding of the current situation, planning data is deduced in regard to strategically determined target data [Fot-2020].

The subsequent **concept planning phase** is based on the actual and planning data. It represents the most important and creative phase of the planning process. According to *Fottner*, this rough planning phase is twofold. Firstly, a structural planning is executed, before a system planning complements the phase[3] [Fot-2020]. The *structural planning* aims at the creation of several structure variants. To do so, the processes are specified in corresponding charts respectively ideal patterns. *Hompel et al.* proposes the development of process variants [Hom-2018, p. 362]. The structural planning determines the functional and organizational units and their interrelations. Further, it delivers an ideal function scheme and the facility's communication concept. All business activities and sequences are defined without having their spatial arrangement regarded. Within the following *system planning*,

[3] As an alternative, a differentiation of this planning stage into rough, fine, and real planning is proposed in other literature, yet the same planning content is covered [VDI-2489].

the resources and material flow systems are selected and dimensioned. The selection of appropriate material handling equipment for the facilitated logistics tasks includes a design of required capacity and area. Based on the ideal function scheme, type and quantity of operating and personnel resources are determined and feasible concepts for material and information flow are developed [VDI-5200, pp. 12–14]. These concepts are understood as system variants, which are reviewed and evaluated within this phase. Concluding the concept planning phase, a rough layout of the selected concept is created [Fot-2020].

The third general planning phase is the **detail planning phase** [Fot-2020]. Within this phase, the fine layout of the planning object is determined including building design, operating resources and calculated costs. The goal is to specify the elements within the aimed-at logistics facility in detail, so that documents required for approval and tender can be prepared. Therefore, the material, information and communication flows are visualized in detail and described as processes. This basically stipulates the allocation of products and resources to the processes, the sequence of process steps, the organizational integration and the working aids used. The user requirement specifications are created[4]. The previously developed rough layout is detailed based on these specifications and thus, transformed into the fine layout. Based on this fine layout statutory approval applications as well as specifications of services are prepared [VDI-5200, pp. 15–17]. Further steps are the contacting of manufacturers and authorities and carrying out the tendering process [Hom-2018, p. 370]. Therefore, various offers are procured and the bidders list is created. The detail planning phase is concluded by an evaluation of the offers and the awarding of contracts to suppliers [Fot-2020].

The core planning process of the logistics facility ends after the detailed planning phase with the preparation of the realization and the approval of the final plans developed by the suppliers [Fot-2020]. The following **implementation phase** is part of the project supervision and management process. The realization is coordinated, monitored, and final project documentation is created. In particular, this refers to the documentation of services actually provided and their associated costs. Deviations from the planning are detected and corrected if necessary. From a planning perspective, this phase is terminated with the provision of evidence that the logistics facility meets the specified performance level and the

[4] Detailed information on user requirement specification is given in DIN 69901-5 or by *VDI* [VDI-3694].

evaluation of the actual resulting facility [VDI-5200, pp. 18–22].

Current Logistics Facility Planning Processes

Numerous planning process approaches are described in literature, mainly referring to *development planning* and *replanning* of logistics facilities. The approaches differ mainly in number of phases and thus, in level of abstraction [Hom-2018, p. 348]. These approaches are common sense process approaches and follow the basic planning principles, such as rough to fine planning, or ideal to real planning [Fei-2020].

Figure 2-23 shows different state-of-the-art planning process approaches based on literature research. They all share the phase character and cover the process from initial definition of objectives through planning stages in different levels of detail up to the realization and ramp-up of the logistics facility. For visualization purpose, the phases of the state-of-the-art approaches were colored according to the general planning phases presented by Figure 2-22. Some of them cover subsequent operation or decommission phases. These are not part of core planning process phases and thus, no color was assigned.

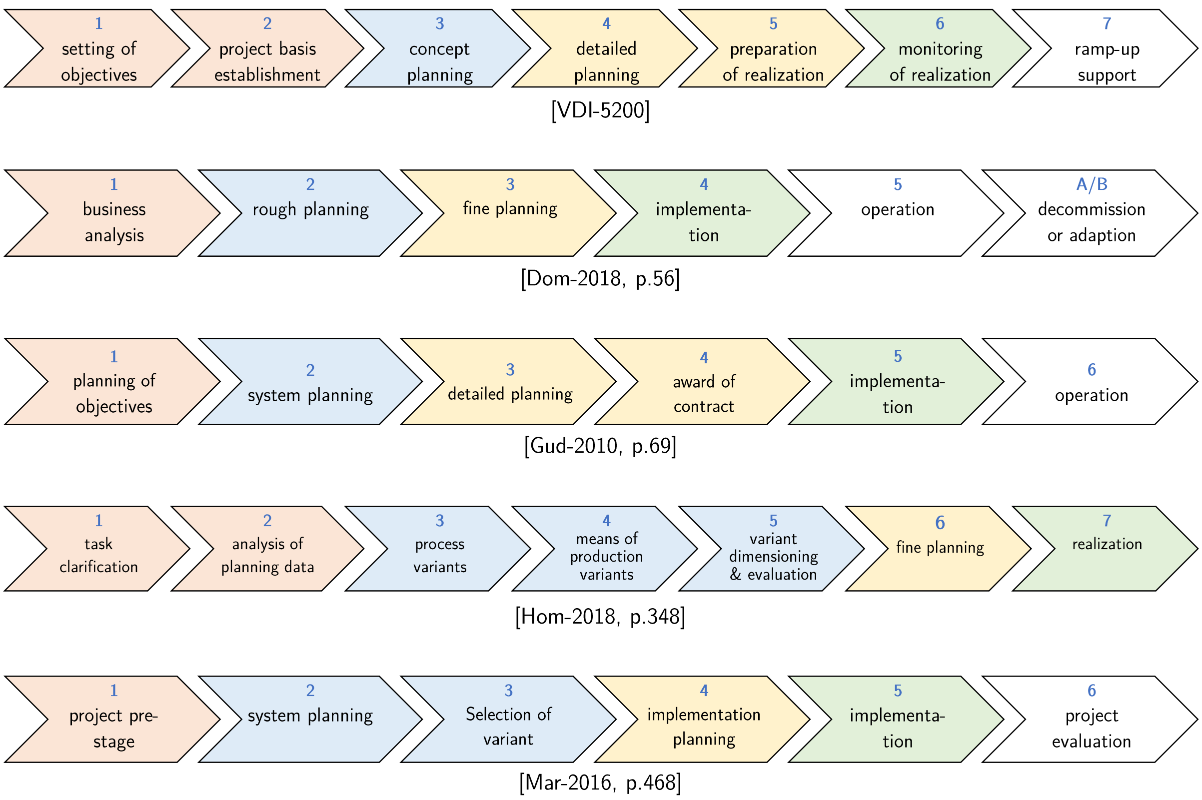


Figure 2-23: State-of-the-art logistics planning processes; color clusters according to the general planning phases, presented in Figure 2-22 [VDI-5200; Dom-2018; Gud-2010; Hom-2018; Mar-2016]

Challenges in Current Logistics Planning

Each of the processes delivers a phase model for contemporary, methodically validated logistics facility planning. However, current logistics planning exhibits certain weaknesses on a process level as well as on an organizational level.

Large project teams, as they are required for complex logistics planning projects, imply a high degree in division of labor. Thus, individual work packages may fail in forming a coherent overall concept. Further, individual project participants may stick to an accustomed divisional thinking instead of keeping the overall concept in mind. Current planning approaches barely stimulate a cross-domain systems thinking mindset among the project participants to cope these issues [Hom-2018, p. 343].

Putting the spot on key planning content, *Durchholz* criticizes a strong focus on technical concepts and realization rather than on logistics processes. In regard to the examination of different possible solutions, the approaches described in literature do only refer the creation of variants to a layout level. Instead, a variant formation is postulated on process level, as only a well-designed process leads to an optimal logistics system. In addition to that, *Durchholz* observes that different ways of processes control, supply principles, or guiding principles such as flow or pulling processes are disregarded to certain extend [Dur-2014, pp. 19–21].

Being established for many years, the planning procedures are still rather document-centered. This led to problems such as misunderstanding and disregarding of interrelations or data [Jet-2007, p. 34]. To avoid such issues, computer support is applied nowadays. Yet, latest digital modeling technologies, such as consistent modeling of digital twins, are barely used in current logistics planning projects, or are only used in very late project phases [Fei-2020]. Instead, networks of numerous individual, encapsulated digital systems are applied. *Dombrowski et al.* state that these arose networks of digital models, methods, and tools still lack in connectivity among the used computer support. Tools used are not linked with each other and data exchange does not work or only works to a very limited extent. Although software interfaces usually exist, an individual adaption for data exchange is often required for each system [Dom-2018, p. 72]. Information systems which are not capable of providing the required data or data interface contribute to an inadequate planning [Hom-2018, p. 343].

From an application perspective, *Durchholz* finds fault in the connection between the theoretical planning process descriptions and the planning progress itself. The abstract and static process approach schemes do not provide best practice guidance in relation to the application of theory. They are claimed to not provide support for both, planning content (what to do) and the corresponding application (how to do) [Dur-2014, p. 26].

In a nutshell, the presented process approaches are well-defined and confirmed by an application in many projects over the last years. However, they are mainly driven by common sense and empirical knowledge. Thus, several challenges exist in terms of managing rising complexity, integration of digital support or giving best practice advice for application of logistics planning processes.

2.3 Summary and Conclusion from Literature Review

In this concluding section of the theoretical baseline, a short summary is given and subsequently the directions from literature review are presented.

Summary

MBSE is a holistic approach unifying the *principles of SE*, namely systems thinking and best practice process models, together with digital *system models*. It mainly aims at an increased efficiency in engineering throughout the whole life cycle and provides a state-of-the-art support for multidisciplinary engineering activities. The core of MBSE is the system model, which represents a pragmatically reduced, abstract image of the engineering object of interest. The system model is developed based on *three collaborating pillars*: the modeling language, tool, and methodology. For each pillar, several implementations exist, from which a *consistent set* has to be selected at the beginning of an MBSE project.

Intermodality denotes the use of multiple modes of transport within one transportation chain. Key concept of intermodality is the *transshipment of load units* among the concerned means of transport without handling the goods themselves. Today, common load units are containers, swap bodies, or semi-trailers. The change in mode of transport is facilitated by **intermodal terminals**. Core part of an intermodal terminal is its *transshipment area*, where the freight is handled and the means of transport exchange their load units.

In contrast, smaller load units (e.g. pallets) are used in **intralogistics freight handling**. Intralogistics basically refers to the in-house organization of material and information flows. Important *intralogistics processes* are the receiving, storage and picking, consolidation and packing, and shipping of load units. These processes are facilitated by the interaction of different *intralogistics system components*, such as conveying, sorting, storage, and order picking systems. Various process approaches to planning of intralogistics systems exist. They share a common phase structure, covering the phases *initial investigation*, *concept planning*, *detail planning*, and *implementation*. Logistics planning processes are in a mature stage, although new challenges concerning among others rising complexity and digitization support arise.

Previous work on the **NGT CARGO logistics terminal** covers the system analysis as part of an initial MBSE approach. Within the previous work, the terminal's system context, stakeholders, requirements, and UC were defined. Final outcomes are a detailed specification of *groups of goods* and the terminal's *top-level system process*. As following project phase, a simulation using Modelica is planned for validation purposes.

Conclusion from Literature Review

From a logistics point of view, the overall NGT CARGO logistics terminal planning project refers to a greenfield planning of a total logistics site, including all corresponding planning domains. In literature, planning domains and levels are described as useful to define the project's extent. Thus, the scope of the modeling activities within this thesis can be adjusted by selecting an *appropriate planning domain respectively level*. In addition, project complexity depends on the object flows of interest, such as material flow, information flow, energy flow, and communication flow.

Reference processes for logistics facility planning exist and were described as established and matured. Yet, several *weaknesses* were revealed in literature research. State-of-the-art logistics planning faces problems of divisional thinking and a premature focusing on obvious technical variants. Great potential for improvement is seen in enhancing interdisciplinary communication and the use of digital models and tools. Yet, consistency and interconnectivity was found as key issues in application of current tools.

Considering these challenges, MBSE *seems to be promising*. MBSE exhibits no domain-specific properties and facilitates multidisciplinary engineering. Focus is laid on top-down creation of system respectively process variants, and comprehensive macro and micro development cycles are provided. The system model enables enhanced stakeholder communication and captures project knowledge as consistent single source of truth. As described in literature, MBSE is spreading out across various disciplines and is about to become a standard approach for complex projects. Consequently, the application of MBSE to a logistics facility planning project stands to reason. It is expected to contribute to a more consistent and comprehensive planning approach.

3 Modeling Approach

This section introduces the applied modeling approach. Firstly, modeling tasks are defined providing a macro structure (Section 3.1) and appropriate implementations of core concepts for system model development are selected (Section 3.2). Based on this, Section 3.3 describes the selected modeling methodologies as well as the applied modeling approach.

3.1 Modeling Tasks

According to the research objectives defined in Section 1.3, the modeling targets the development of an architecture for the NGT CARGO logistics terminal core. To provide a modeling structure and to clarify the wordings, four modeling tasks were defined, delivering terminology as it was applied within this thesis.

1. Determine a **base architecture**

   A base architecture describes the system from a structural view at the project starting point. It equals an initial framework as a basis for the development process and considers preset decisions regarding technologies and requirements [Wei-2014, p. 52]. In order to sketch the initial architecture frame of the NGT CARGO logistics terminal, a base architecture has to be determined.

2. Develop a **functional architecture**

   The entirety of a system's functions is a technology-independent system description. The second task aims to the exploration of a functional entirety defining the process properties of the NGT CARGO logistics terminal core. In the thesis, functional architecture is specified as *'[...] the set of functions and their sub-functions that defines the transformation of input flows into output flows performed by the system to achieve its mission'* [SEB-2020].

3. Derive a **logical architecture**

   Architecture in general allocates functions to elements, which are arranged in a certain structure [Hab-2019, p. 157]. In case of the logical architecture, these elements and their relationships embody fundamental logical concepts or properties of the system in its environment [ISO-42010]. In the terminology of this thesis, the logical architecture is regarded as a generic and abstract description of the conceptual structure, supporting the logical operations of intralogistics freight handling within the NGT CARGO logistics terminal core.

4. Identify a **product architecture**
   The product architecture is the most concrete description of the system. Finding a possible product architecture means to identify an arrangement of real physical components providing a specific design solution in order to satisfy the logical elements and requirements [Wei-2016b, p. 59]. Since there is no unique solution for a product architecture of a system, the final modeling task targets the derivation of one exemplary possible structure variant.

These tasks describe the main artifacts to be created within the modeling approach. Hence, the next step is to select appropriate core concepts for the system model development.

3.2 Applied Core Concepts

Based on the findings from the state of the art (see Subsection 2.2.1), suitable implementations for the core concepts of system model development were selected (Figure 3-1).

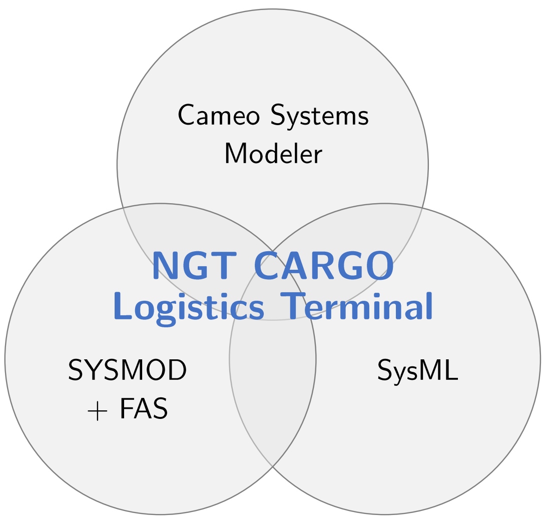


Figure 3-1: Applied core concepts for system model development (own illustration)

As modeling language, it was decided to use **SysML** for the modeling purpose within this thesis. SysML is capable of supporting the specification, analysis, design, verification and validation of complex systems and thus, it is suitable for the purpose of this thesis [Obj-2019]. Apart from being the most popular modeling language for MBSE, SysML was chosen for two further reasons. The first major point was, that the initial system model set up by DLR in previous work was developed using SysML (See Subsection 2.1.4). Approaches towards a transformation of existing models from one modeling language in another are described in literature. Yet, this transformation is a critical task and is associated with major effort [Bad-2018]. The second main reason for choosing SysML as modeling language was the availability of support for the acquisition of language skills. For SysML, dedicated lectures were available at Technical University of Munich [Bra-2020; Wal-2018]. Furthermore, lot of support exists in literature [Alt-2012; Wei-2014; Fri-2014].

As a modeling tool, the **Cameo Systems Modeler** was selected. According to its vendor NoMagic, this tool is capable of implementing all aspects of systems in the most standard-compliant SysML diagrams [NoM-2021]. Supporting literature exists [Cas-2017]. Apart from this, the Cameo Systems modeler is easy to learn and quite popular among MBSE users [Clo-2010, p. 121; Piv-2021]. As with the language, the usage of this tool for previous work by DLR was a major point since no transformation of the existing base model to another tool was needed. Furthermore, support and license were provided by DLR. The Cameo Systems Modeler's ability for simulation and solving parametric models by using Modelica is a benefit in regard to the following validation of the model planned by DLR (see Subsection 2.1.4) [NoM-2021; Ehr-2020].

Based on literature review, it was decided to follow a twofold MBSE methodology approach, combining **SYSMOD** and **FAS**. Both are methodologies explicitly targeting product development and they are well documented, as literature is available and easily accessible [Wei-2016b; Wei-2014; Lam-2014]. In addition, they are mature enough as proved by various application examples in industry projects [Dae-2014; Clo-2010, p. 127]. The appropriateness for use together with SysML and Cameo Systems Modeler is given [Wei-2014; Wei-2016c]. Regarding the content, SYSMOD provides methods and processes for modeling the base, logical, and product architecture based on an initial system behavior analysis. Since functional architecture is spared out in SYSMOD, the FAS is a fitting complement. It guides the passage from the system's behavior to its structure by providing the functional architecture. As all artifacts required by the modeling tasks are delivered, the selected methodologies are sufficient for the purpose of this thesis.

Together, the selected implementations facilitate the development of a system model for the terminal architecture in accordance with the modeling tasks.

3.3 Tailoring of Modeling Methodologies

The MBSE methodologies *SYSMOD* and *FAS* are selected to guide the modeling of the terminal system architecture. Yet, a customization is necessary to adapt the generic methodologies to the specific needs of this thesis.

The intention of the following subsections is not to provide a comprehensive description of the methodologies. Concerning this matter consultation of related literature is recommended [Lam-2014; Wei-2016b]. Instead, a short overview with focus on the relevant key concepts is given for SYSMOD (Subsection 3.3.1) and FAS (Subsection 3.3.2). The tailored modeling approach combining both, SYSMOD and FAS, is presented in Subsection 3.3.3.

3.3.1 SYSMOD

SYSMOD is a best practices toolbox, that can be regarded as collection of methods rather than a process description. Various *SYSMOD methods* and corresponding artifacts (so called *SYSMOD products*) are specified by this methodology. During the modeling, appropriate SYSMOD methods are applied to create the SYSMOD products contributing to the model. The SYSMOD methodology itself does not explicitly prescribe which methods to use in which sequence. The order and collection of applied methods strongly depends on modeling focus, the system's domain and vary from project to project. Iterations and loops in execution of methods are recommended [Wei-2016b, pp. 1–11].

The most relevant methods for addressing the stated modeling tasks within this thesis are presented by Figure 3-2. Each method and its artifacts are described in detail by *Weilkiens* and propositions about how to execute the corresponding tasks are given [Wei-2016b].

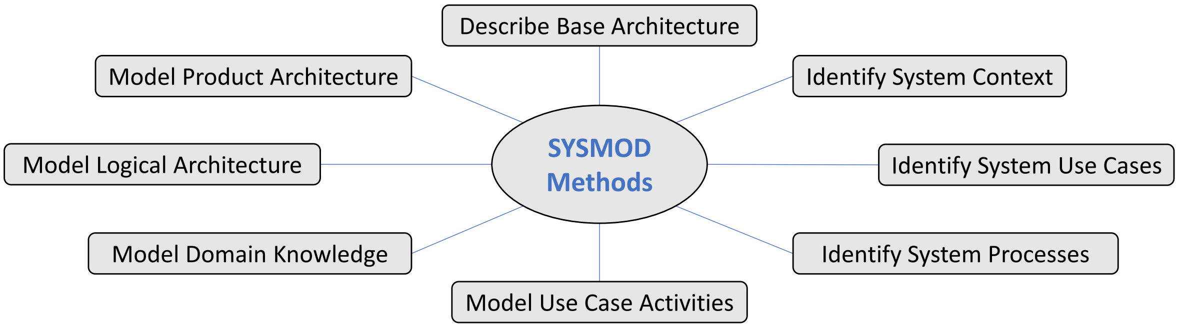


Figure 3-2: SYSMOD methods with key relevance for this thesis (own illustration based on Weilkiens *[Wei-2016b, p. 11])*

However, to provide orientation for a potential SYSMOD approach, *Weilkiens* introduces one possible logical order of execution of the SYSMOD methods. The resulting process contains certain SYSMOD methods and suggests a logical sequence [Wei-2016c, pp. 5–9]. A comprehensive example of a different SYSMOD process is described in literature [Wei-2014]. It is important to emphasize, that these processes propose a logical order rather than a timely order. The execution of the iterative methods may overlap.

3.3.2 FAS

The SYSMOD methodology does not provide a particular method for the modeling of a functional architecture which is postulated by modeling task 2. To address this problem the FAS methodology provides a well-defined procedure how to develop the functional architecture [Lam-2014]. Since SYSMOD and FAS overlap to a certain degree, only the distinct process steps of FAS as applied in this thesis are introduced in the following.

The core of the FAS methodology refers to four main steps. The artifacts resulting from these steps are depicted by Figure 3-3. The basis for the FAS methodology are certain system artifacts provided by SYSMOD, mainly the UC and corresponding essential activities (0). Firstly, for each UC an **activity tree** is created containing the essential activities as blocks (1). Key step of the FAS methodology is to rearrange the essential activities from the system's overall activity tree into so-called **functional groups** (2). These artifacts represent the functions of the system. Although the definition of functional groups is supported by heuristics, common sense and comprehensive system expertise is required. Next, **functional elements** are traced from functional groups as a transition from behavior-oriented

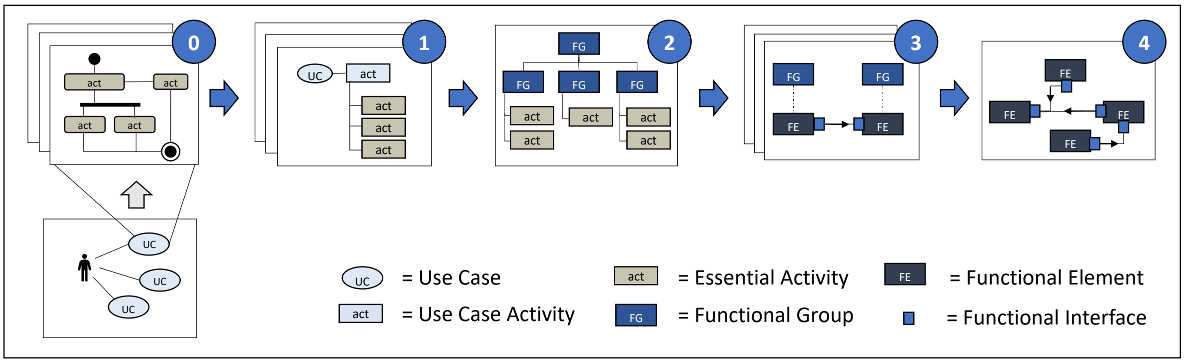


Figure 3-3: Main artifacts within FAS: use case activities (0), activity trees (1), functional groups (2), functional elements and interfaces (3), and functional architecture (4) (own illustration)

to structure-oriented modeling. Further, the use of functional elements decouples the life cycle of UC from the functional architecture and implements a change control. The **functional interfaces** among functional elements are derived from the inputs and outputs of the essential activities (3). Finally, the flows among functional elements respectively their functional interfaces are realized, resulting in the **functional architecture** (4) [Lam-2014; Wei-2016c, pp. 189–229]. Within this thesis, the functional architecture is used to facilitate further architecture development in SYSMOD.

3.3.3 Tailored Modeling Approach

Based on selected SYSMOD methods, the resulting SYSMOD artifacts, and the propositions of possible SYSMOD processes in literature, a tailored modeling approach was developed. It provides a basic frame for modeling with the FAS methodology applied as embedded activity. The resulting approach is illustrated in Figure 3-4.

Analogous to the SYSMOD processes described in literature, the depicted procedure states the logical sequence of applied methods, their inputs and resulting artifacts. No rigid time-related sequence is implied due to its iterative character. The artifacts developed within previous work by DLR were considered as inputs (requirements, stakeholders, top-level system process). Corresponding to the modeling tasks the four types of architecture were regarded as outputs of the project. Apart from the SYSMOD and FAS methods, logistics-specific activities were added to realize the tailoring to the intralogistics terminal context (specification of basic intralogistics process, logistics toolbox).

The resulting procedure is capable of guiding the development of the terminal system architecture. It concludes this section as customized, domain-specific modeling approach applied in this thesis.

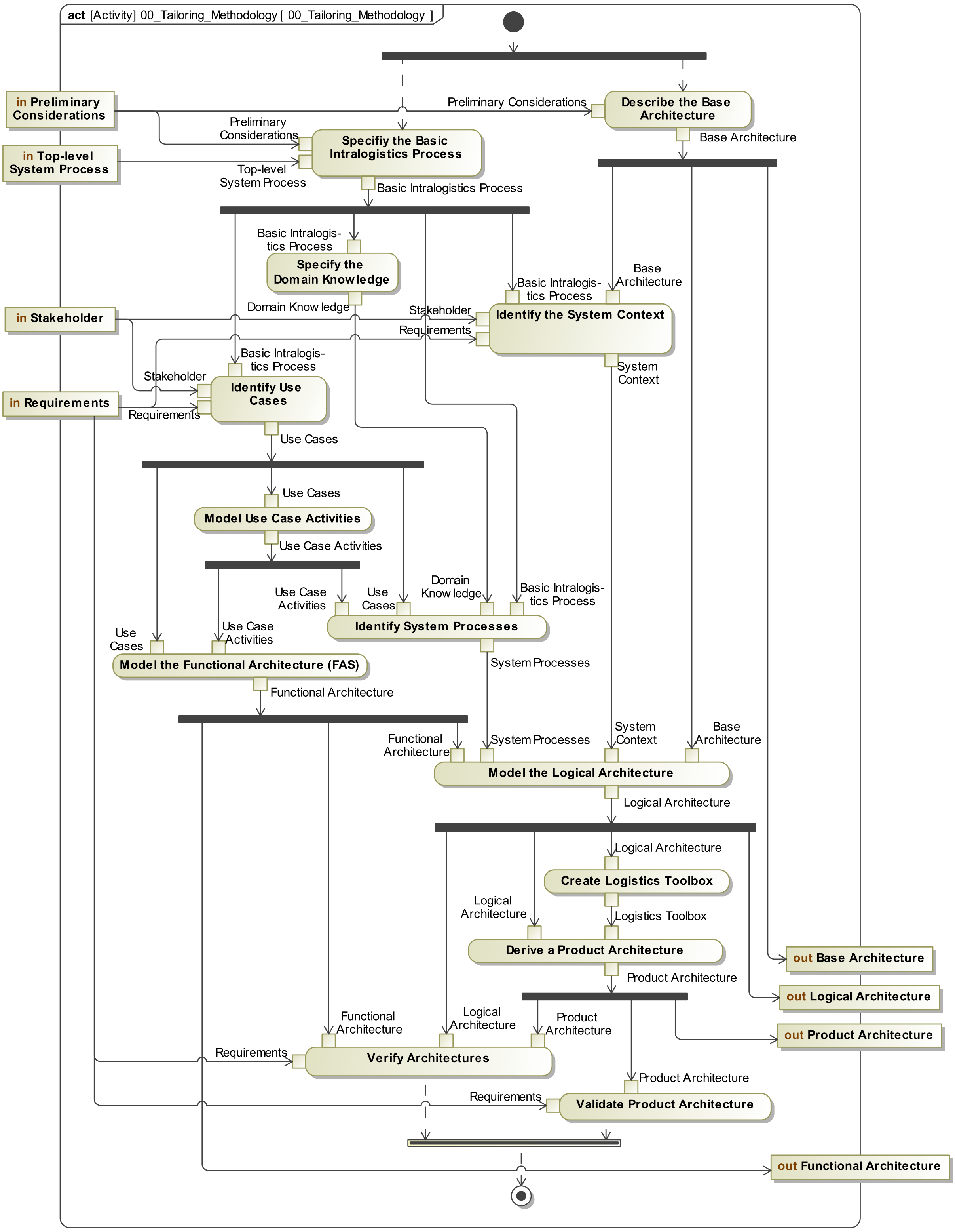


Figure 3-4: Tailored modeling approach as applied within this thesis, based on SYSMOD (own illustration)

4 Development of the Terminal System Architecture

The following chapter describes the development of the terminal system architecture based on the previously introduced modeling approach (see Figure 3-4).

Prefacing, Section 4.1 states the modeling scope and gives preliminary remarks. As first step, Section 4.2 describes the definition of the base architecture and the fundamental intralogistics process. The specification of the terminal's behavior was done within a comprehensive system analysis as described in Section 4.3. The following architecture modeling process is threefold. Section 4.4 covers the application of the FAS methodology leading to the functional architecture. As a next step the development of the logical architecture is presented (Section 4.5), before Section 4.6 concludes with the description of the logistics toolbox and one possible product architecture as a detailed variant of the terminal core.

4.1 Preliminary Considerations and Remarks on SysML

In prior to the modeling, preliminary considerations on the intended level of complexity and relevant system characteristics have to be made. Further, remarks on SysML are given.

Preliminary Considerations

The object to specify is the NGT CARGO logistics terminal within a certain degree of detail and by the use of models. Due to reasons of scope, the logistics planning domains location and external infrastructure were not covered by this thesis and the planning level was lowered down to the transshipment building only. Thus, on-site infrastructure, corresponding external interfaces, or vehicle operations and management were not regarded. For the purpose of the thesis, the term *'terminal core'* was defined and concerns the intralogistics transshipment section facilitating freight exchange among different means of transport. Although they are required from a construction's perspective, building systems and structure, energy supply, and office systems were not covered within this thesis either.

The involved means of transportation were limited to the NGT CARGO itself and the group of *road vehicles*. This group was defined including conventional trucks, sprinters, cars, or the Next Generation Car (NGC) Cargo Mover[1]. Other possible means of transport mentioned by the NGT CARGO concept, such as planes, drones, ropeways, or cargo bikes were disregarded. Transshipment directions of interest were Rail-Rail, Rail-Road, and Road-Rail. Apart from transshipment with temporary storage within the terminal core, crossdocking was covered as direct transshipment operation without storage for all transshipment directions. For reasons of scope, order picking and customs were regarded as external services and their consideration was reduced to the provision of theoretical process interfaces.

Within the terminal core, the focus was laid on the material flow. The physical flowing unit is a Load Unit (LU). Since the NGT CARGO concept considers pallets and ULD as load units to transport, LU refers to these load units. A further decomposition of LU, e.g. for storage, picking, or intralogistics handling, was excluded. Information flows were only partial included. All types of information were gathered and denoted as Load Unit Information (LU Info). Thus, the flowing unit for information is LU Info. Being out of scope, any information processing or management was disregarded, and data infrastructure was reduced to the provision of minimum connecting interfaces when necessary. Information flows to external services or systems were ignored.

Remarks on SysML

For the modeling, SysML was used as modeling language. To tailor the generic SysML language to the problem-specific domain, stereotypes were used based on the SYSMOD profile given by *Weilkiens* [Wei-2016b, pp. 163–180]. Whenever necessary reading advice is given so that no initial SysML knowledge should be required to understand the key messages of the views. An overview about SysML elements and the SYSMOD profile is given in Appendix B. For a comprehensive description of syntax and semantics of SysML reference to the relevant literature is recommended [Fri-2014; Wei-2014].

The modeling steps and results are illustrated using SysML diagrams from the model. They show a particular view of the system architecture or other artifacts specified by the system model. These views show the model artifacts with relevance to the corresponding issue to display. Although the described procedure seems to follow a sequential order, the architecture development was characterized by iterations and numerous changes. The following views represent the latest status of the corresponding SysML system model.

[1] The NGC Cargo Mover is a future modular freight vehicle concept developed by DLR [Wei-2016d].

4.2 Base Architecture and Basic Intralogistics Process

As a first step in the modeling process, the base architecture was defined with respect to the stated preliminary considerations. The base architecture determines the level of abstraction and frames the system specification process and therefore the scope of entire modeling activities. All boundary conditions and other prescribed structures are considered in this architecture concept. Within this thesis, the base architecture was found especially valuable to illustrate the basic system idea and to establish a common understanding of the system of interest (SOI) at an early stage in the project. This common understanding includes main interfaces and the basic structure of the SOI.

Figure 4-1 illustrates the **base architecture of the logistics terminal** with focus on its transshipment area. The terminal core was defined as SOI, as indicated by the stereotype «system of interest».

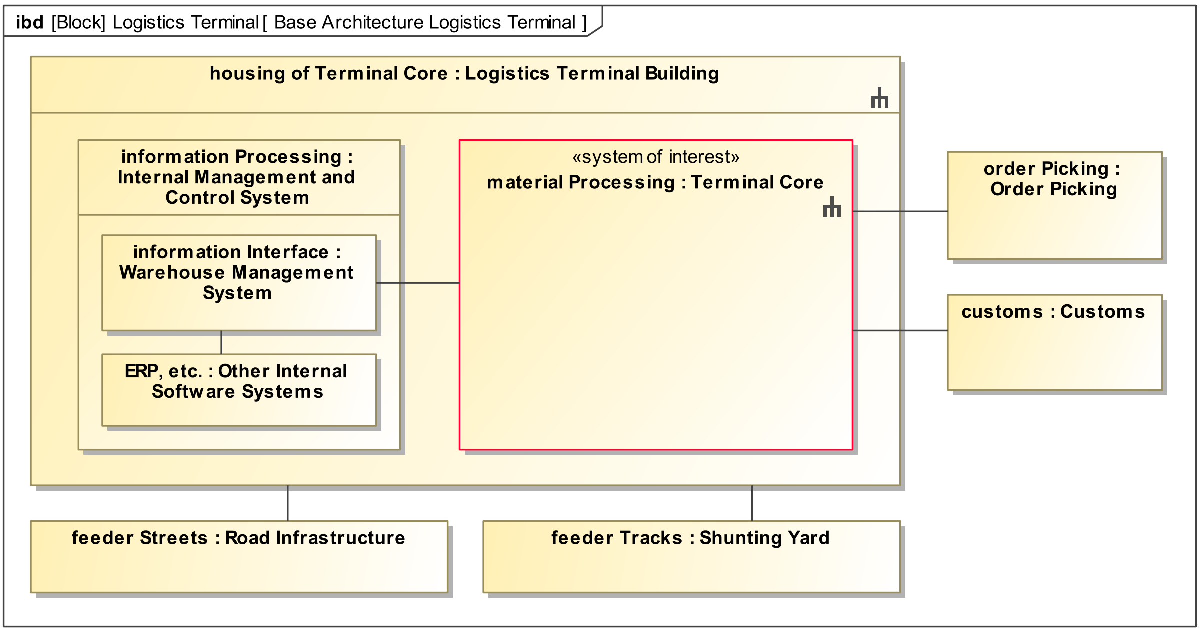


Figure 4-1: Base architecture of the logistics terminal with terminal core highlighted as SOI (SysML view)

Focus of the view displayed in this figure was laid on the prescribed structures around the terminal core as object to model. The terminal core was defined as being located within a logistics terminal building. This building sets the frame for the terminal core and contains all necessary building equipment which was excluded from scope before (See Section 4.1).

The building further provides interfaces towards the surrounding infrastructure for the NGT CARGO (shunting yard) and the road vehicles (road infrastructure).

The terminal core was associated to the role[2] of material processing. Several interfaces towards other systems were implied, without further specification about numbers or location. A key interface was implemented to the internal management and control system in the role of information processing. To keep this information interface simple, the warehouse management system (WMS) was defined as unique interacting information system forwarding any information to the corresponding internal software system when necessary. Further interfaces to the external services order picking and customs were planned. Due to their prescribed external character, these services were displayed outside of the logistics terminal building. This was done for clarification reason and does not imply a decision about the actual location of these services.

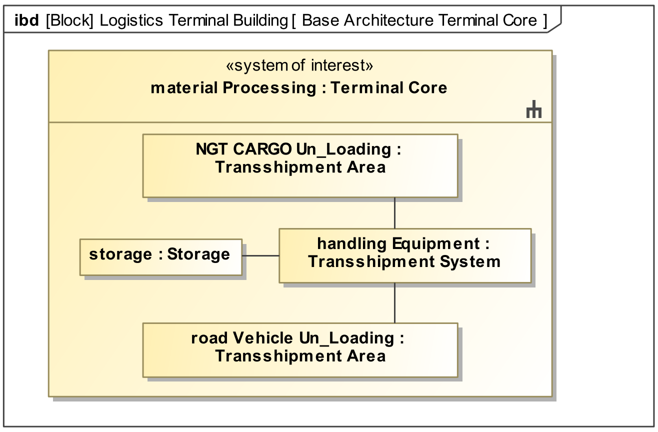


Figure 4-2: Base architecture of the terminal core, SOI (SysML view)

Within the previous view in Figure 4-1, the terminal core itself is treated as black box without internal structure. To give an idea of the basic understanding of the SOI, Figure 4-2 reveals the **base architecture of terminal core** as an insight into the SOI block. At the beginning of the system specification, the terminal core was understood as system providing two separated transshipment areas, where either the NGT CARGO railcar or the road vehicle are unloaded respectively loaded. These two locations were linked by a transshipment system, where the freight was supposed to be transshipped using handling equipment. Finally, a storage was planned to buffer time shift between unloading and loading of freight.

In parallel to the base architecture development, the **basic intralogistics process** of the terminal core was sketched. As a counterpart to the SOI's structure specified by the base architecture, the basic process definition targets the initially intended behavior of the SOI.

[2] SysML notation of element headers: '*role of element (in context of view)* : *type of element*' [Bra-2020].

The purpose of defining a basic intralogistics process at this point in time was to sharpen and clarify the covered logistics activities in an early stage of model development. Based on common sense and literature review, the basic intralogistics transshipment process was specified as illustrated by Figure 4-3.

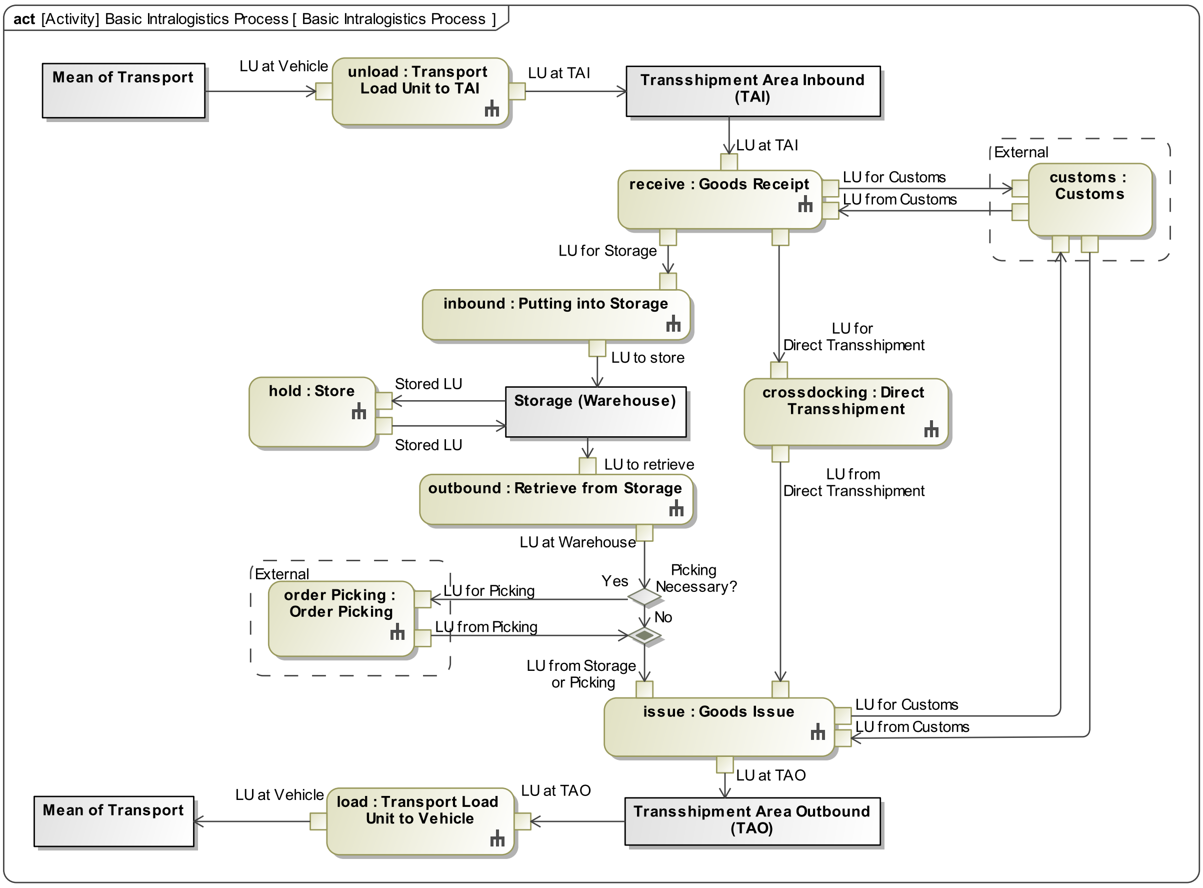


Figure 4-3: Basic intralogistics process facilitated by the terminal core, material flow only (SysML view)

It was defined that the terminal core process starts/ends as soon as the transshipped LU is at the door of the corresponding mean of transport. In between the transshipment shall be done, facilitated by the base architecture introduced previously. The view displayed by Figure 4-3 shows actions and basic spatial allocations[3]. Initially, the LU is transported from the door of a vehicle (NGT CARGO railcar or road vehicle) to a transshipment area inbound (TAI) where the goods receipt process takes place. In case of crossdocking, the LU is directly transshipped and runs through the goods issue at the transshipment area outbound (TAO). In the other case, the LU is put into storage, stored, and retrieved, before it runs through the goods issue process at TAO as well. The interface for external order

[3] From theoretical SysML perspective, Figure 4-3 is partly incorrect, since central buffer nodes were used as it is not intended by language semantics. Yet, this way of visualization was considered useful to comprehensibly demonstrate the coherence of actions and locations. It was used for illustration purpose only.

picking was coupled to the storage retrieving process to disjoint the LU as late as possible. Process interfaces for customs were established within the goods receipt (inbound) and goods issue (outbound).

In the process specification, the transshipment areas from the base architecture evolved from simple vehicle handling areas to load handling and buffer areas. The reason to install buffering areas in a logical order between the means of transport and the terminal core itself was that the vehicle unloading and loading process is time critical. This is especially of interest for the outbound direction, as otherwise the time advantage of an intended fast handling would be lost due to long conveying times of the LU from the storage to the vehicle [Hom-2018, p. 332].

Further, in contrast to the base architecture, the basic intralogistics process already implied a spatially separation of inbound and outbound flows for the interface towards the vehicles. According to *Hompel et al.*, separated material flows are especially beneficial in case of high transport volume. Overlapping is prevented and a material flow free of interference is guaranteed [Hom-2018, p. 321]. With respect to these benefits, the implementation of TAI and TAO is justified.

However, as the terminal's behavior is basically similar for rail and road transshipment, no differentiation among the mean of transport was made on a process level.

4.3 System Analysis of the Terminal Core

Once the basic structure and process of the terminal core were defined, various steps were taken to approach the SOI's behavior systematically and thus, to further specify its processes. These steps can be summarized as system analysis. This system analysis covers specification of the domain knowledge (Subsection 4.3.1) as well as the development of the system context (Subsection 4.3.2). To provide a comprehensive picture, the development of UC and the system process are gathered in Subsection 4.3.3, before the derivation of UC activities as a top-down specification of the system process is presented (Subsection 4.3.4).

4.3.1 Domain Knowledge

Within this project, the domain knowledge was used to define key domain objects occurring in the object flows of the model. It can be regarded as common baseline to communicate a universally valid understanding of these objects and thus, is presented first. The domain model was continually updated and grew with ongoing project progress.

In this thesis, material flow and information flow were regarded. As stated in Section 4.1 the corresponding flowing objects were *LU* and *LU Info*. Figure 4-4 shows the **domain blocks of the relevant objects** as a view extracted from the domain knowledge.

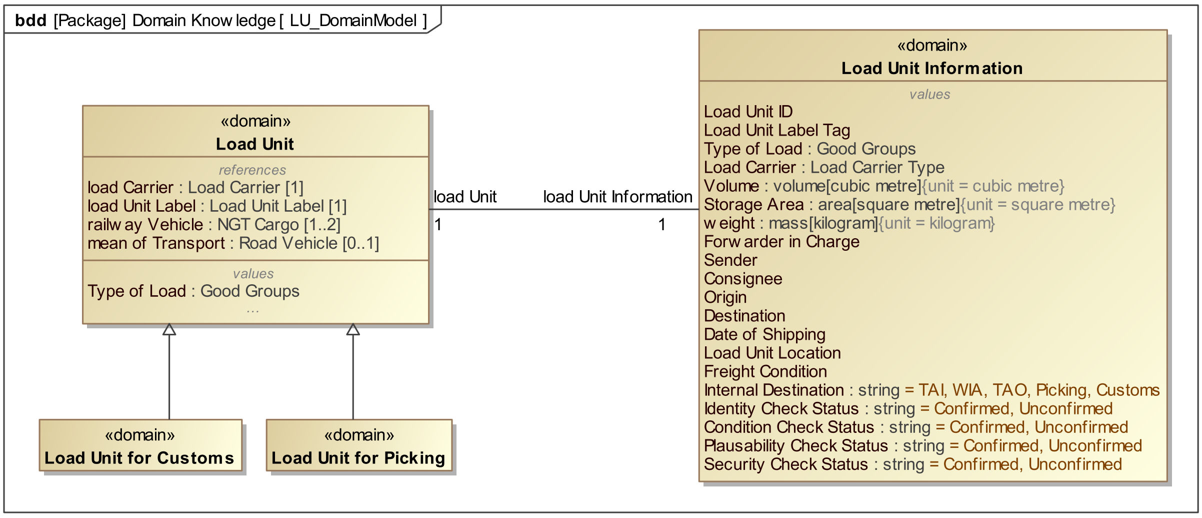


Figure 4-4: Extract from domain knowledge with focus on flowing objects as domain blocks (SysML view)

The *LU* was described in the domain knowledge as independent domain block with several properties. References to the domain blocks of *load carrier* or *LU label* were defined (not displayed). Further, the LU refers to one or two *NGT CARGO rail vehicles* and one or no *road vehicle* (multiplicity depends on the different transshipment possibilities). To explicitly distinguish between the LU to transship and the LU to/from external services, further specializations of the general LU were defined using the SysML generalization relationship. The general *Load Unit* denotes the LU as it is transshipped, while the domain blocks *Load Unit for Customs* and *Load Unit for Picking* were introduced as variants. The variants inherited all features from the general LU, yet they were used in different context.

In addition, the LU domain block was associated with the domain block LU Info. This domain block gathers all information required within the transshipping process. The information was implemented as values of the LU Info domain block. Different types of information were listed, such as general and shipping information, physical values, LU location, internal destination, or check status information. This set of information is partly comparable to the accompanying document information, although the list is not conclusive.

It is clear that this treatment of information is a major simplification. However, due to the limits in project scope and the corresponding focus on material flow, it was accepted.

4.3.2 System Context

The system context illustrates the SOI in its environment and states the relationships to the actors[4]. As described in the modeling procedure (Figure 3-4), several inputs were needed to specify the system context. Apart from the base architecture and the basic intralogistics process, this refers to the identification of requirements and actors.

The **requirements and actors** were mostly provided by the previous work on the terminal by DLR. Since scope was reduced to the terminal core, the input was tailored accordingly. To enhance readability, the list of selected requirements of the terminal core is presented within Appendix C. These requirements were used to define the SOI's interactions with the environment. This facilitated the selection of the relevant actors.

Based on the stated inputs, the **system context** was modeled as depicted by Figure 4-5. It

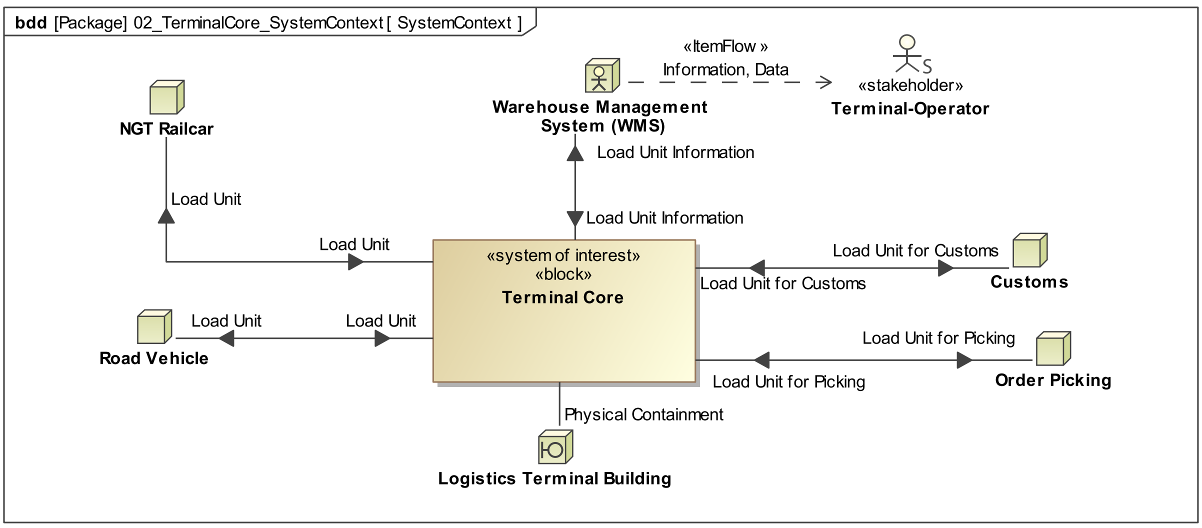


Figure 4-5: System context of the terminal core showing involved actors and relationships (SysML view)

[4] Within this thesis, the term 'actor' denotes both, stakeholders and systems, actively interacting with the SOI.

shows the terminal core surrounded by the selected actors. The *NGT railcar* and the *road vehicle* were identified as key systems interacting with the terminal core by exchanging LU. Further material flows were visualized towards the external systems *customs* and *order picking*. As specified within the domain knowledge (Subsection 4.3.1), the exchanged items are variants of LU. Regarding the information flow, the *WMS* was defined as single interacting system for LU Info. The *WMS* is displayed as user system, since it was seen as key user interface for the stakeholder *terminal-operator*. Finally, the *logistics terminal building* was listed. Although there are no active flows[5], the enclosing terminal building was mentioned as interacting system in the system context to symbolize the physical containment of the terminal core. It was not part of further architecture specification.

4.3.3 Use Cases and System Process

In this thesis, a UC was understood as a specific service of the system under investigation demanded by at least one actor [Bra-2020]. Various services were identified based on the requirements, the selected actors and the basic intralogistics process. These services were consolidated in **six different UC** (Figure 4-6).

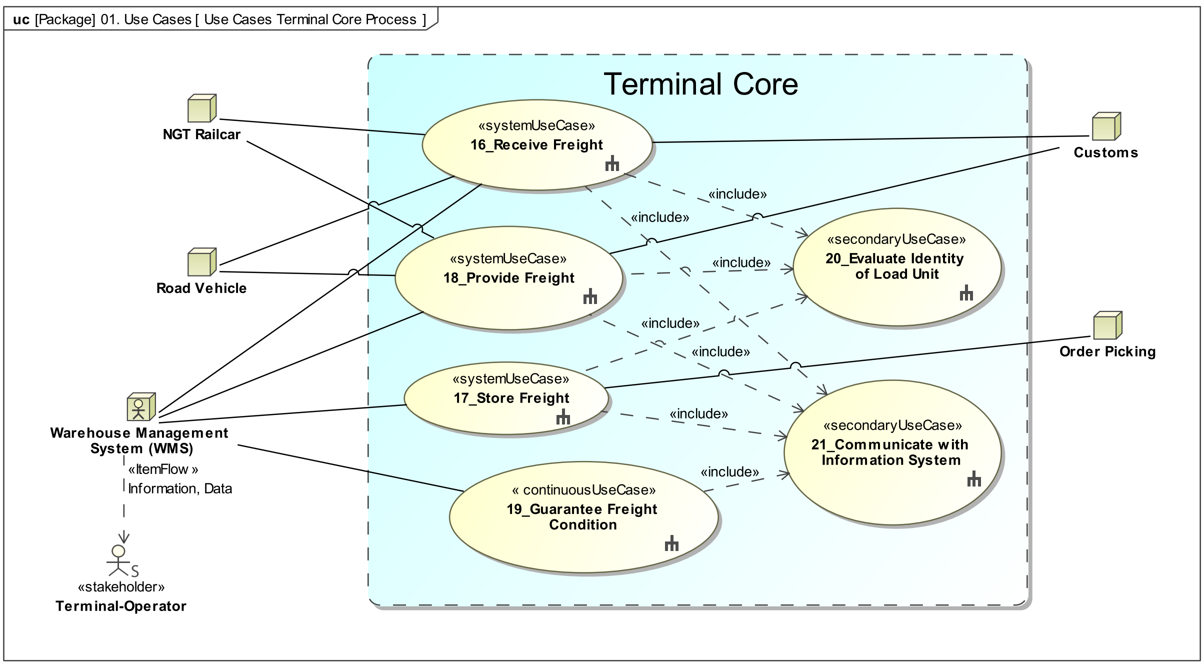


Figure 4-6: Use cases of the terminal core (SysML view)

[5] Refers to information and material flows as stated in the preliminary considerations (Section 4.1).

According to the SYSMOD profile of SysML, a further differentiation of the UC was done using specific stereotypes. The three UC classified as «systemUseCase», namely *receive freight*, *store freight* and *provide freight*, cover the key system services in regard to freight handling. The «continuousUseCase» *guarantee freight condition* represents a service which is constantly provided by the terminal core and refers to the particular environment condition requirements of freight. Finally, *evaluate identity of load unit* and *communicate with information system* were classified as «secondaryUseCase». A secondary UC is a fragment of a UC without an own actor. Instead, the secondary UC represents repetitive sub-services which are included in other UC. Using secondary UC, redundant description of the sub-services was avoided [Wei-2014].

The UC define the terminal's behavior triggered by actors. To further specify the actually executed processes, SysML activity diagrams are used. Since a UC must not be displayed in an activity diagram, a formal transition from UC to activities is necessary [Wei-2014]. Thus, each UC was allocated to a same named **UC activity**. Following a top-down approach, the UC activities were initially treated as black box processes with particular inputs and outputs. These black boxes were arranged in a logical sequence and the inputs and outputs were connected by object flows. The resulting sequence is the description of the top-level **terminal core process** (Figure 4-7). For reasons of readability, only material flows are visualized in the view depicted by Figure 4-7.

In SysML, object flows (solid lines) symbolize the transmission of objects (material or information flow), while control flows (dashed lines) are used to create the logical sequence of activities [Fri-2014]. This difference is important to read the following diagrams correctly.

Analogously to the basic intralogistics process, the LU to transship runs through a sequential key process, represented by the UC activities *receive freight*, *store freight* and *provide freight*. However, the UC analysis revealed that a freight condition management is needed apart from the key transshipment process. Thus, the corresponding UC activity *guarantee freight conditions* is logically parallel-connected to the main material flow. By definition, the remaining secondary UC (*evaluate identity of load unit* and *communicate with information system*) are located on a lower level of abstraction and thus, they were not displayed within this view of the terminal core process.

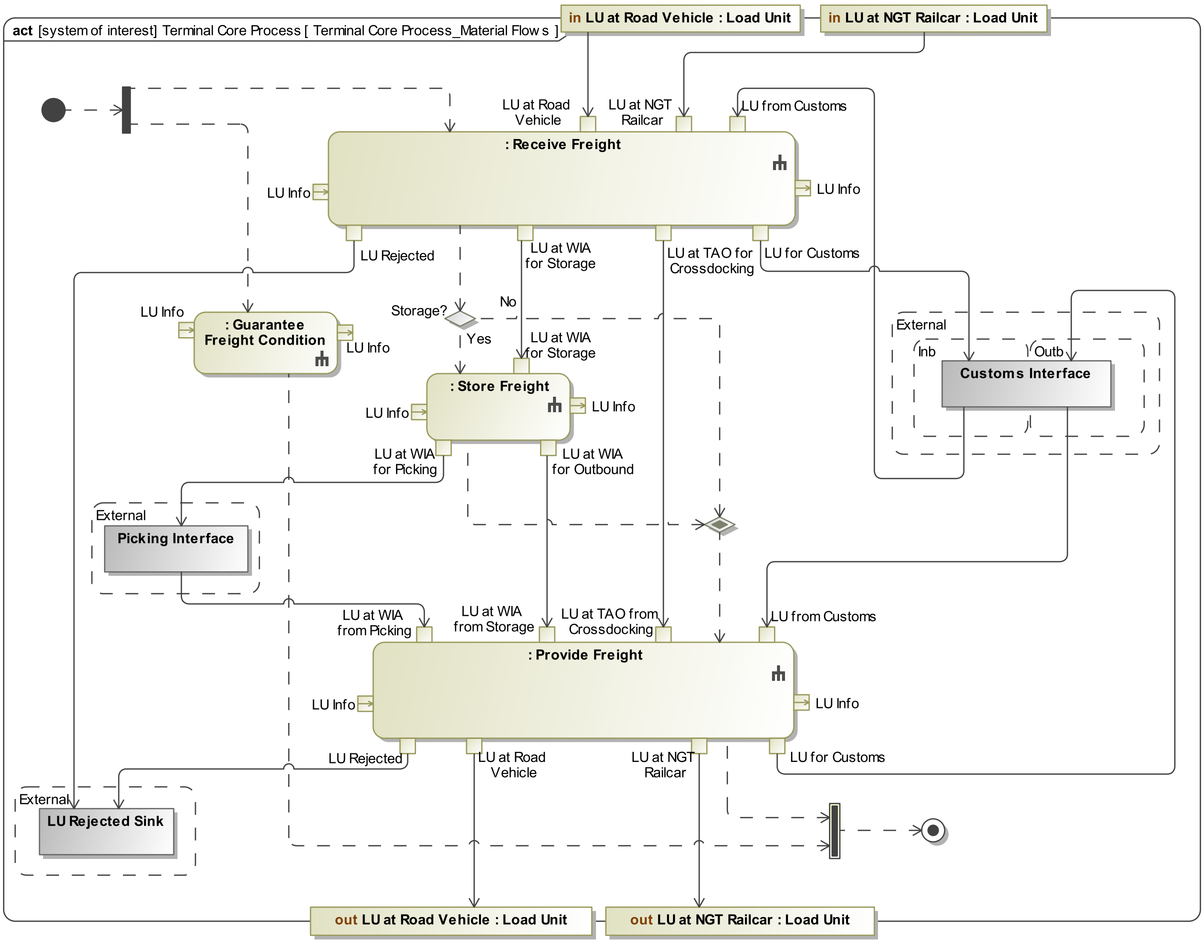


Figure 4-7: Terminal core process, consisting out of use case activities; material flow only (SysML view)

At this stage of specification, the external services *customs* (inbound and outbound) and *picking* were reduced to their corresponding interfaces. Within the terminal core process view, these interfaces were realized as external black box sinks respectively sources.

As a simplification, no further processing or management of damaged, unidentified, or incorrect LU was considered. Thus, a particular *LU rejected sink* was introduced, absorbing all LU which are rejected within the process.

The terminal core process depicted in Figure 4-7 can be regarded as systematic abstraction of the initially defined basic intralogistics process (see Figure 4-3). From a system model hierarchy perspective, the new terminal core process description equals a zoom into the *terminal core process*-action node embedded in the top-level system process of the whole terminal (see Figure 2-21).

4.3.4 Use Case Activities

The UC activities were specified on a next level by defining a **sequence of subordinated essential activities**. In the following, the resulting activity diagram for each UC are presented. To support the readability of the diagrams, the object flows are separated in heading direction. Material flows are headed vertically (top to bottom) and the information flows are headed horizontally (left to right).

Receive Freight

The UC activity of *receive freight* is illustrated by Figure 4-8. It basically covers the process from the pick-up of the LU at the vehicle's door up to its transportation to a further internal destination. In a first step, the LU is conveyed to the TAI where it is buffered until

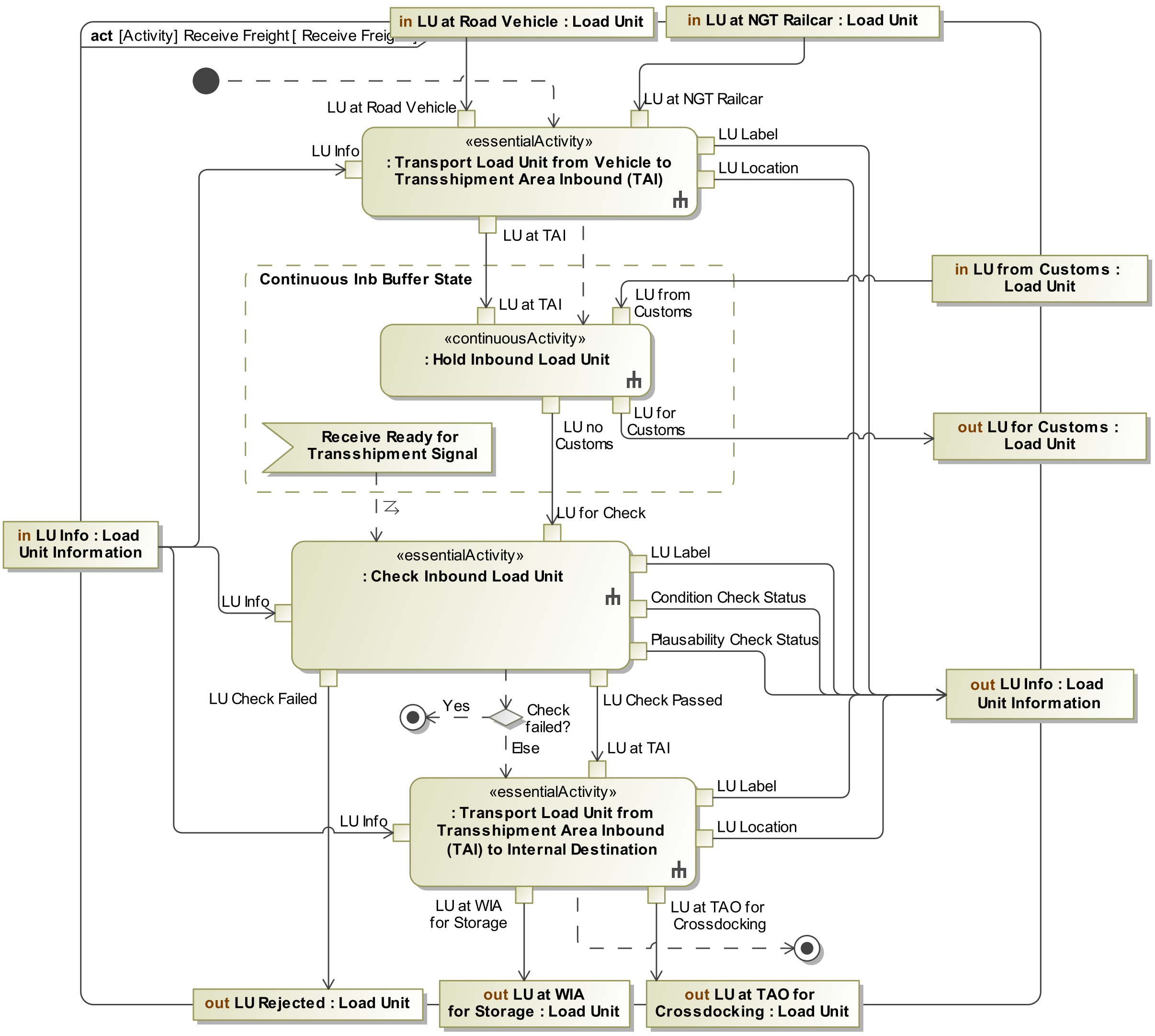


Figure 4-8: Process sequence of the use case activity 'receive freight' (SysML view)

further transshipment is possible. At this point the inbound customs interface is provided. As soon as the signal for further processing is received, inbound checks for quality and quantity of freight follow up. In case of successful checks the LU is transported to the next internal destination, which is either the warehouse interface area (WIA) (storing) or the TAO (crossdocking). If at least one inbound check was failed, the LU is rejected immediately.

The freight handling processes are facilitated by information support from the WMS (LU Info). During the *receive freight* process, the label, the location, and check results of the LU are evaluated respectively determined. The information is used for decision making, documentation, and tracking of LU. Certain information is sent back to the WMS as an update of LU Info[6].

Store Freight

The UC activity for *store freight* covers all activities associated to warehouse operations. The corresponding activity diagram is shown by Figure 4-9. The interface area between the

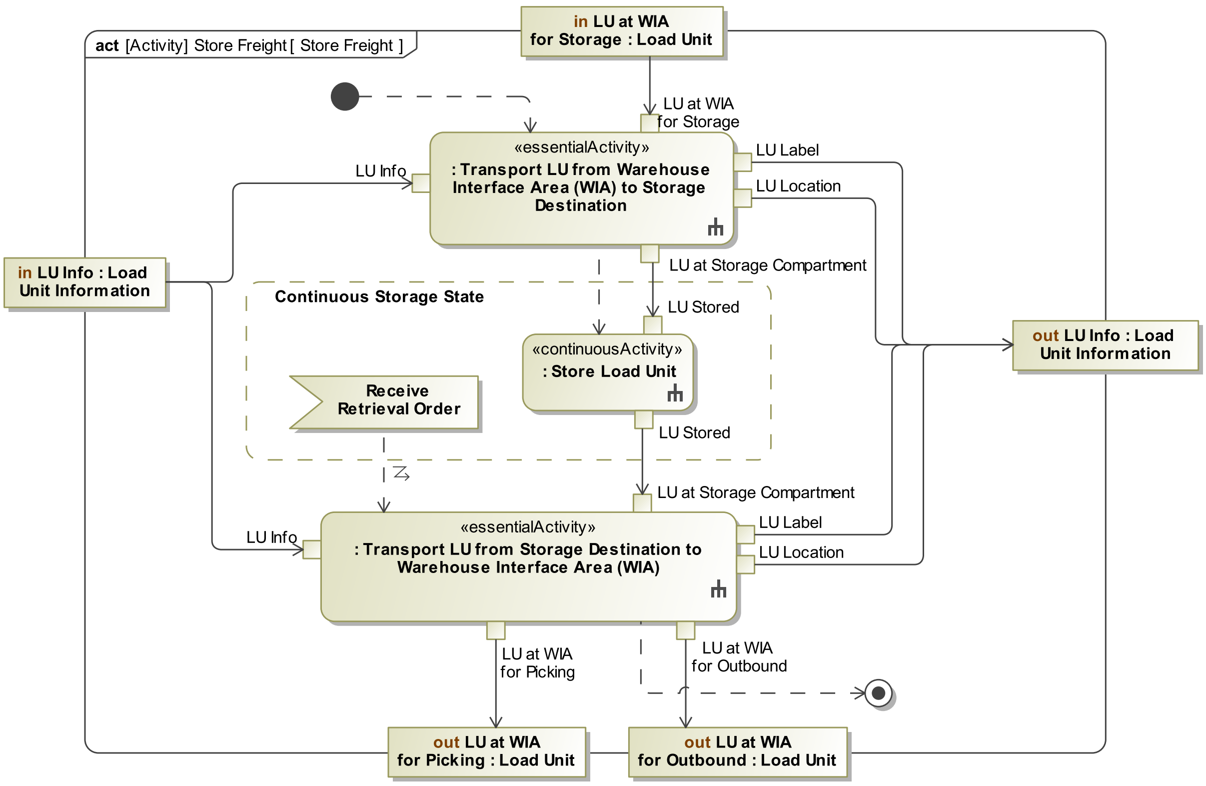


Figure 4-9: Process sequence of the use case activity 'store freight' (SysML view)

[6] In the SysML model, all information outputs were defined as *'role : Load Unit Information'* (type refers to domain block *LU Info*) and summarized in a single output parameter node. To describe the transferred information, each pin was provided with a dedicated role within the activity diagrams.

intralogistics conveying system and the warehouse is denoted as WIA. Here, the inbound LU coming from the *receive freight* activity is handed over to the warehouse conveying system and brought to the reserved storage compartment. Subsequently, the LU enters the continuous storage state. Once the retrieval order is received, the retrieval process takes place. Concluding the *store freight* activity, the LU arrives at the WIA for further transshipment. At this location, a potential interface to the picking system was considered. If picking is requested, the LU is sent to the picking system. Else, the LU is handed over to the *provide freight* process.

Provide Freight

The UC activity *provide freight*, which is presented by Figure 4-10, considers several initial

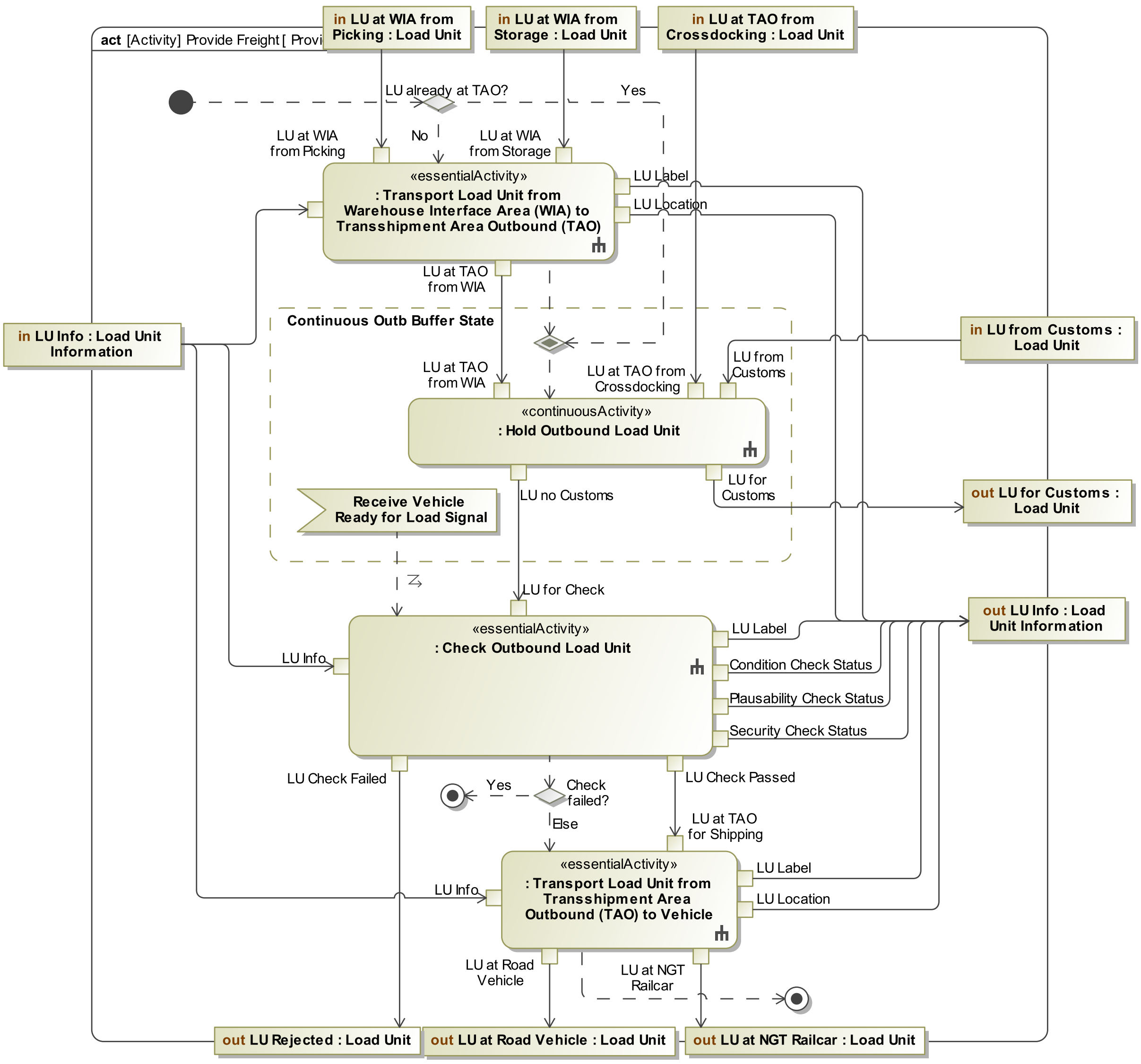


Figure 4-10: Process sequence of the use case activity 'provide freight' (SysML view)

locations of the LU to process. If the LU comes either from the storage or from the picking process, the LU is provided at the WIA. Consequently, the first activity covered by the UC *provide freight* is the transportation of the LU from the WIA to the TAO. In case of a crossdocking procedure the LU is already at the TAO (brought by the final conveying activity in the *receive freight* process). Once the LU is in the TAO, it is buffered until the vehicle is ready for loading. Analogously to the inbound procedure, the continuous buffer state at the TAO serves as the process interface for customs operations. Before the LU is finally transshipped to the vehicle, a final outbound check is passed. In addition to condition and quantity checks, an outbound security check was introduced for LU heading for the NGT railcar. Comparable to air freight transport, the high-speed rail freight was considered as vulnerable mean of transport so that only clean LU are loaded[7].

The positioning of the outbound checks at the very late point in the logical sequence of the provide freight process was done intentionally. From an organizational perspective, it would be better to execute the checks earlier (e.g. before entering the TAO). If a LU fails one check, enough time would be available to correct or manage the error and to channel in the LU again. On the other hand, the outbound checks for the loading in the NGT railcars contain the security check. In air freight transshipment, which was taken as reference for the NGT CARGO security checks, the freight has to be protected from any unauthorized interference after the security check until loading [Eur-2015]. In order to minimize potential hazards after the security checks, their logical position was set as late as possible.

Guarantee Freight Condition

Figure 4-11 gives an insight into the UC activity *guarantee freight condition*. This activity is executed continuously and aims at providing adequate environment conditions required by the goods. The activity was designed with regard to controlled and uncontrolled (permanent) conditions. To guarantee controllable conditions a continuous control loop was introduced. By monitoring and controlling of temperature, humidity, and illumination the corresponding requirements of these conditions shall be fulfilled. The control loop can be left by a stop signal. In parallel, permanent activities executed by building systems, such as restriction of access and protection from fire, electric magnetic radiation, or environmental elements, belong to this UC.

[7] Concepts of known consignors exist for air freight [Eur-2015], yet they were not considered in this thesis.

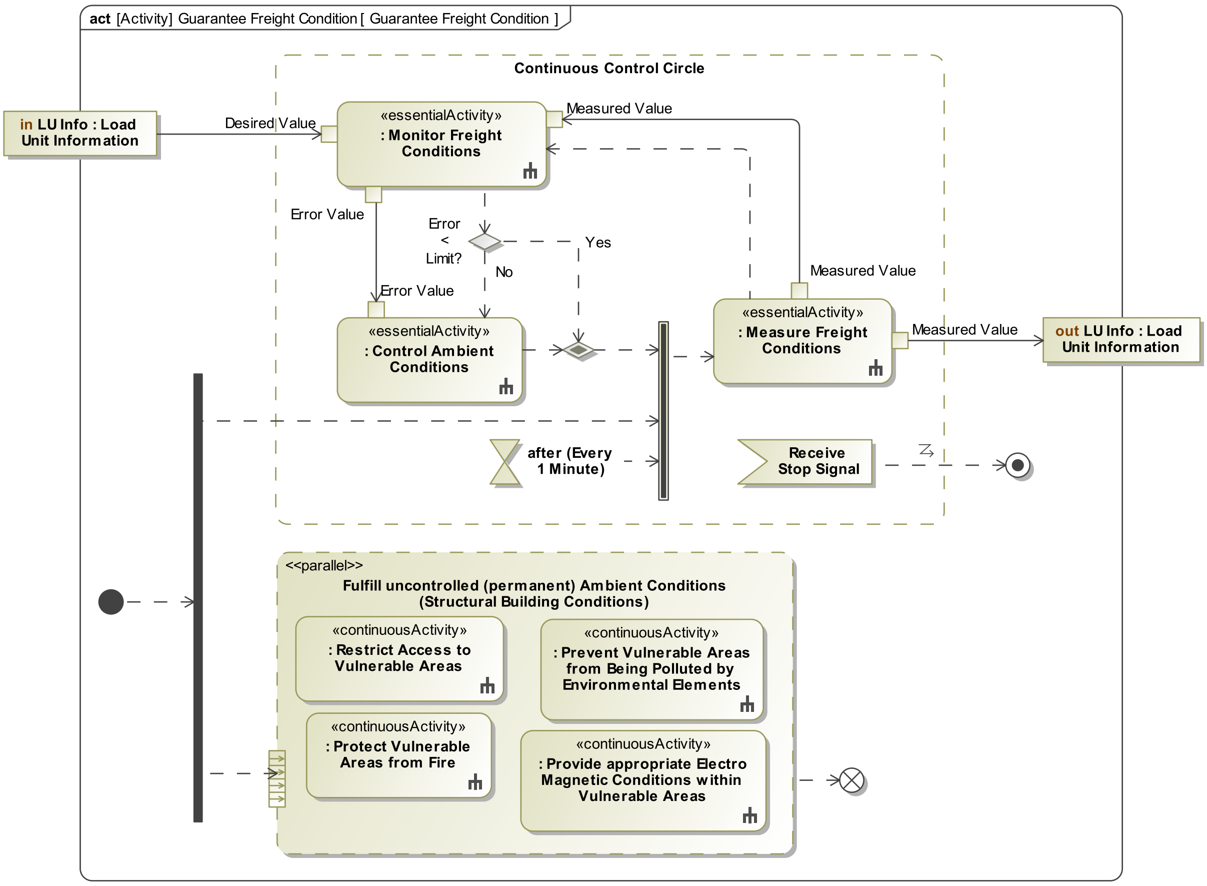


Figure 4-11: Process sequence of the use case activity 'guarantee freight condition' (SysML view)

It is important to emphasize, that this UC strongly depends on the transshipped good. Different goods may require different conditions or control values.

Evaluate Identity of Load Unit

The identification of the LU is a repetitive process, which is executed various times within the previously introduced UC activities. As an example, at the beginning of the essential activity *check inbound load unit* (part of the *receive freight* activity, see Figure 4-8) the LU has to be identified. Consequently, the UC *evaluate identity of load unit* is part of the *check inbound load unit*-activity on a more detailed level.

Therefore, this UC was designed as secondary UC. Its corresponding UC activity is displayed by Figure 4-12. It contains a single activity, where the label of the LU to identify is evaluated. This activity can be regarded as the interface between material and information flow. The LU label itself serves as identification tag of the particular LU. The information output of the activity is the LU label as type LU Info.

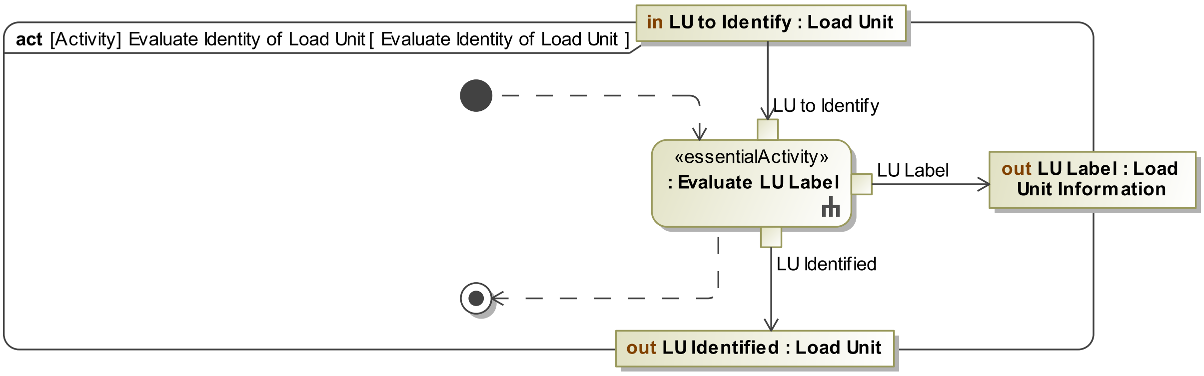


Figure 4-12: Description of the use case activity 'evaluate identity of LU' (SysML view)

However, this activity does not specify the degree of information integration in regard to the label[8]. As information flows and management was not in the focus of this thesis, this was intentionally kept rather generic.

Communicate with Information System

Analogously to the previously mentioned secondary UC, the UC activity *communicate with information system* was applied on a lower level of detail within other UC activities. The following activity enables the exchange of information with the WMS interface. Consequently, every other activity dealing with exchange of data requires the service provided by this secondary UC activity. As shown by Figure 4-13, the UC activity differentiates in sending and receiving of information.

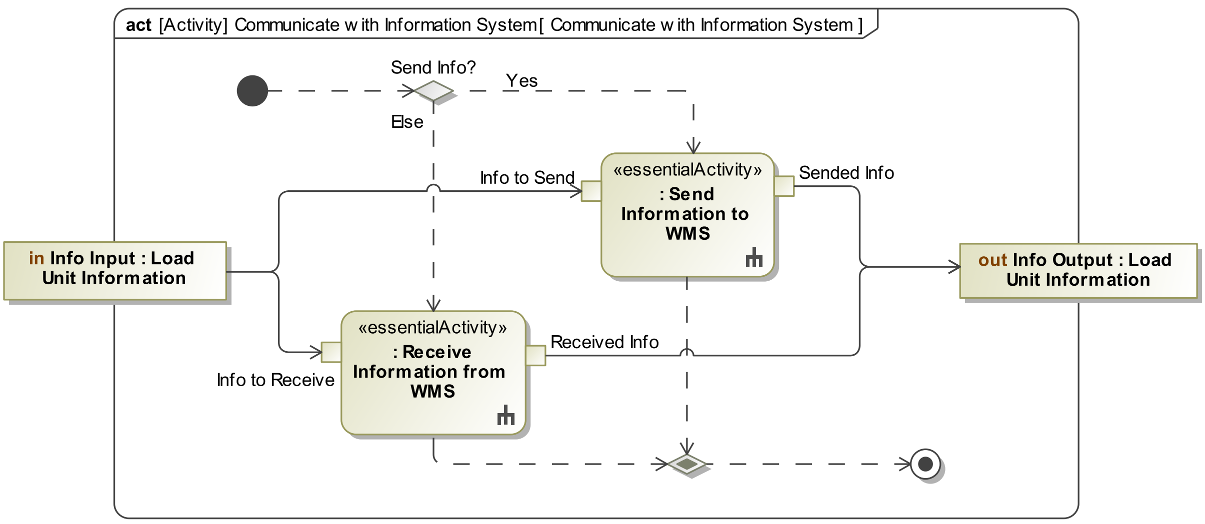


Figure 4-13: Description of the use case activity 'communicate with information system' (SysML view)

[8] Examples for degrees of information integration are ID-on-tag, data-on-tag, or agent-on-tag. Refer to literature for more information [Net-2010].

Interim Conclusion on System Analysis

At this point, the specification of the terminal core's behavior has already reached a very detailed level. Thus, a few aspects on the progress so far shall be mentioned.

The system's behavior bases on the initially defined basic intralogistics process. This guiding course of actions was formalized using UC based on the actors' demands. These UC were transformed into activities and put into a logical sequence. Particular inputs and outputs enabled the definition of material and (on a basic level) information flows. In the last section, this sequence of UC activities was further specified by essential activities, taking the process on a next level of detail.

It is important to notice, that this zoom-in procedure can be continued nearly to infinity, continuously increasing accuracy without ever reaching a level of 100% accuracy. On the other side, for each additional level of detail, the complexity of the model as well as the required domain knowledge increase. To complete the project within the given frame of time, an appropriate balance between accuracy and complexity had to be chosen.

Some of the essential activities were even specified on an additional level in the model. Yet, the terminal core's behavior described by the essential activities presented before was judged to be mature enough to proceed with the structural specification. Therefore, development of a comprehensive further level of behavioral detail was refrained from.

4.4 Functional Architecture

In a next step, the essential activities have to be transformed into functions that describe the main tasks of the terminal. The interactions of these functions yield the **functional architecture**, which is therefore a modeling artifact referring to the system structure [Wei-2016c, p. 202]. To derive the SOI's functional architecture based on the determined UC activities, the FAS methodology was applied.

Firstly, the SOI's **global activity tree** was created (Figure 4-14). An activity tree is static representation of a top-level activity and its subordinated activities. An activity tree shows call hierarchies and no ownership hierarchies. Control flows and thus, sequences of activities are not displayed due to its focus on structure [Wei-2014].

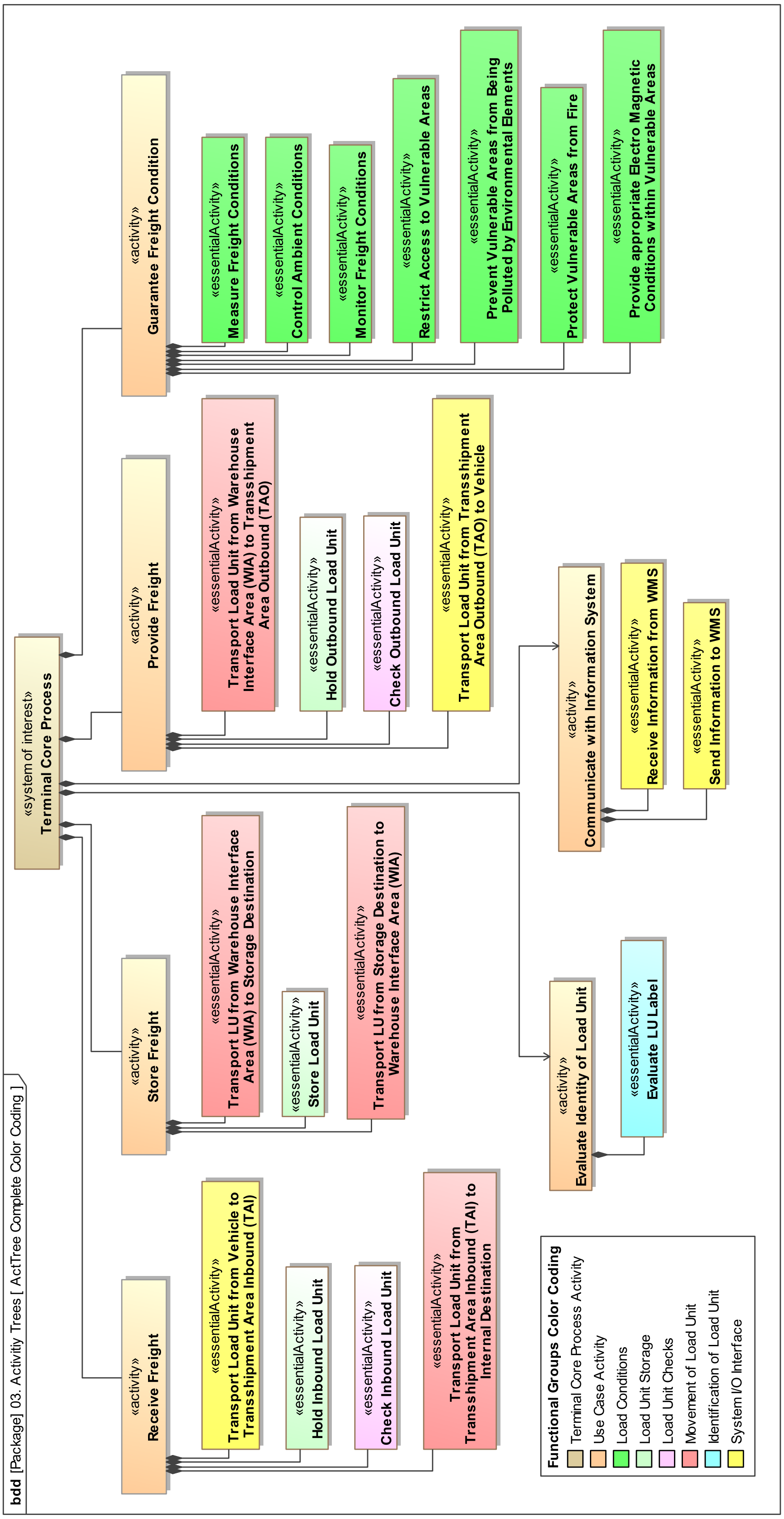


Figure 4-14: Global activity tree with grouping of activities by means of use cases; color coding according to functional groups (SysML view)

At the top-level of the global activity tree, the *terminal core process* is set as SOI. On the second level, the six previously introduced UC activities are displayed. Beneath each UC activity the called essential activities are listed. Consequently, this view shows all activities occurring within the terminal core process up to the third level and grouped by the means of UC. The coloring of the nodes within this view already anticipates the allocation to functional groups, which is the key outcome of the next step in the FAS methodology.

Within this next step, **functional groups** were defined and the essential activities were allocated and regrouped according to these functional groups.

In context of FAS, literature defines functional groups as *'set of strongly-related [...] activities'* [Lam-2014]. Therefore, the main task within the second step of FAS was to cluster the existing activities into independent groups with highest possible functional cohesion. This referred especially to related or identical objectives, inputs or outputs and aimed at low external complexity and high internal functional complexity. Apart from common sense, the definition of the functional groups was facilitated by heuristics, given in literature [Wei-2016c, pp. 199–202; Lam-2014]. Examples of applied heuristics are the grouping according to interface operations, abstract and secondary UC, and shared data or objects. Related functional groups were gathered within a superordinate functional group. The resulting nine functional groups are illustrated by Figure 4-15.

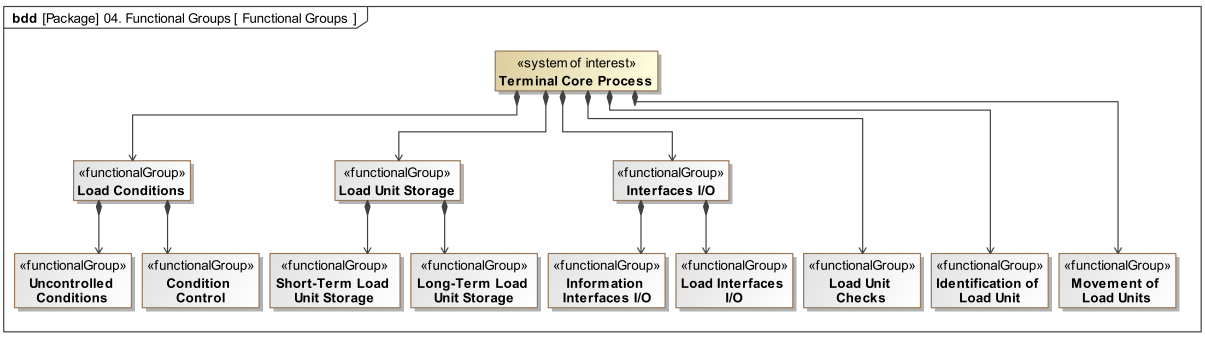


Figure 4-15: Functional Groups (SysML view)

Once the functional groups were defined, every essential activity displayed in Figure 4-14 was traced to the corresponding functional group. This is trace relationship is visualized additionally by the color coding within the view (which was anticipated previously). Subsequently, the tree was restructured. The UC activity nodes were replaced functional group nodes, and the activities were regrouped by mean of functional group affiliation. The resulting **functional group tree** is displayed by Figure 4-16.

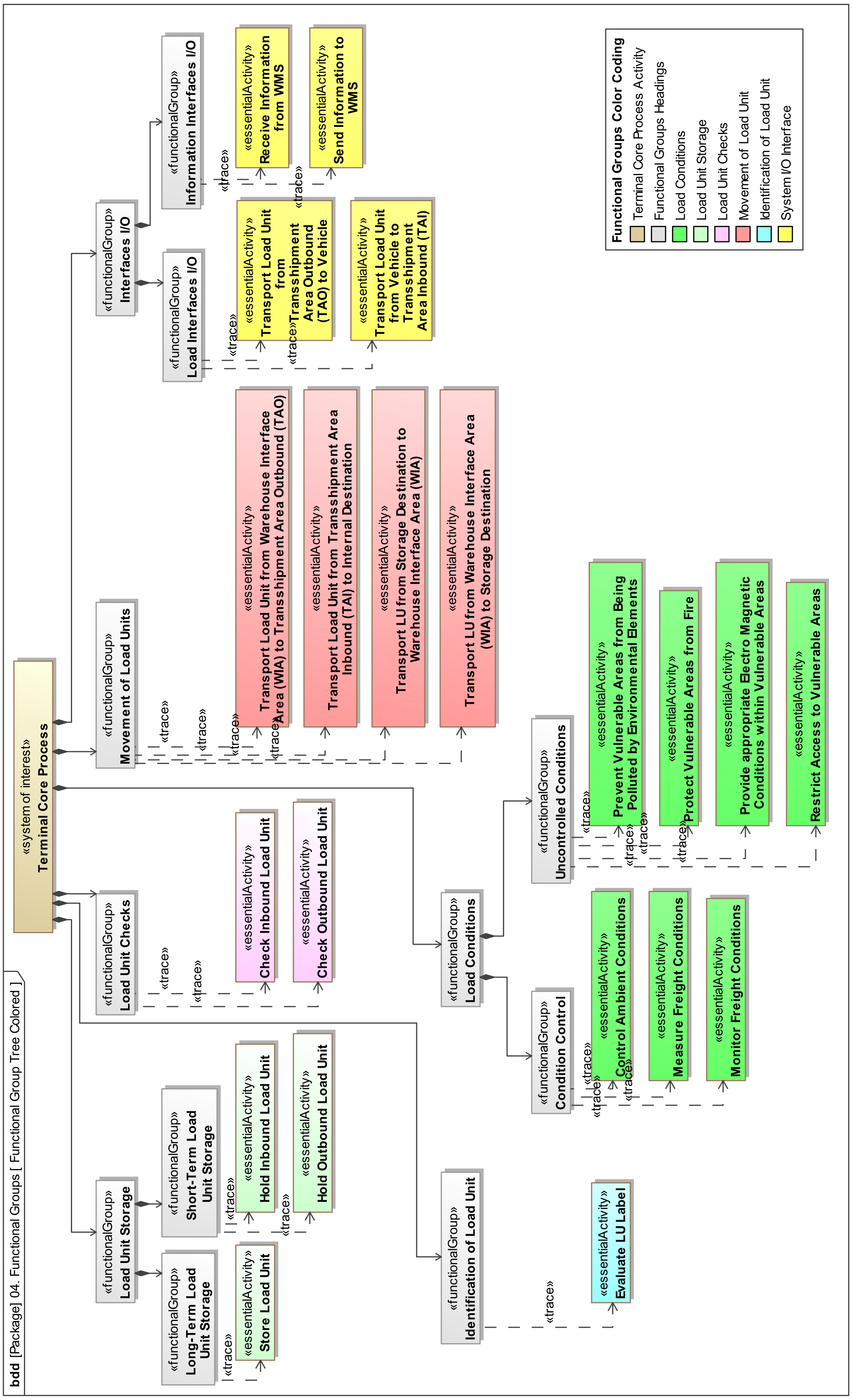


Figure 4-16: Functional group tree with grouping of activities by means of functional groups (SysML view)

As mentioned initially, the functional architecture refers to structure rather than behavior. Formally, the functional groups still describe the terminal core's behavior. The following steps within the FAS procedure addresses this transition from behavioral to structural elements and the composition of the terminal core's functional architecture.

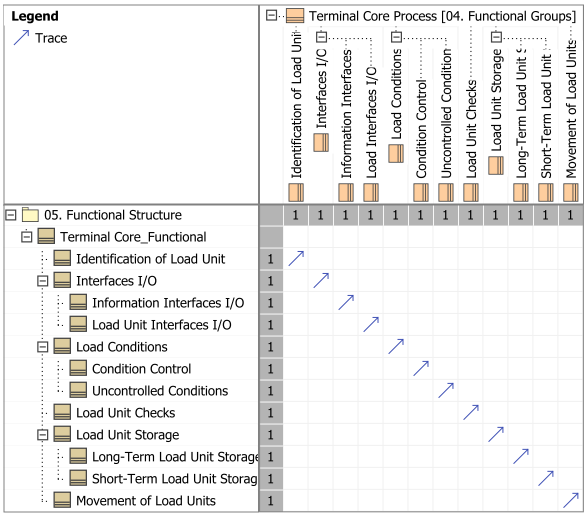


Figure 4-17: Mapping of functional elements to functional groups (SysML view)

Therefore, the functional groups were expressed with a same named functional element and connected with a one-to-one trace relationship (Figure 4-17). In contrast to a functional group, a **functional element** is an abstract structural system component. It defines the relation between one or more inputs and outputs by means of a function [Lam-2014]. In the SYSMOD/FAS profile a functional element is modeled as «functionalBlock» exhibiting the properties of a SysML block. Due to the one-to-one tracing, the framework of functional groups was equal to the framework of functional elements except for one difference: the top-level node was changed from the *terminal core process* (behavior, specified by functional groups) to the *terminal core* itself (structure, specified by functional elements). Collectively, the entity of functional elements builds the **functional structure** of the terminal core.

In contrast to the hierarchy-oriented functional structure, the functional architecture lays additional focus on functional interrelations and flows. In the FAS methodology, these interrelations are denoted as functional interfaces [Lam-2014]. In SysML, interfaces of blocks are facilitated by ports. A port may exhibit a flow property, so that it is capable of mapping the block's inflow or outflow of a specific item type. Making use of the block properties of functional elements, **functional ports** were defined and allocated to these elements. By the configuration of the functional ports as proxy ports with flow properties, the input and output flows of *LU*, *LU Info*, and *LU for customs* respectively *picking* were modeled. Finally, the **functional interfaces** were realized by modeling connecting object flows among the ports of corresponding functional elements.

As final step of the FAS methodology, all functional elements were visualized within an internal block diagram of the terminal core. By displaying the ports and functional interrelations a connected structure was formed. Furthermore, ports were allocated to the terminal core itself as the SOI's top-level functional interfaces towards external systems. This included a *loading interface* to the vehicles, an *information interface* to the WMS, and a *customs* respectively *picking interface* to the corresponding external service. For a better understanding of context, these external systems were visualized as blocks in the view, too.

The resulting **functional architecture** is illustrated in Figure 4-18. To enhance readability, the object flows are colored according to the type of conveyed item (*LU Info*, *LU*, and *LU for customs* respectively *picking*). The arrows indicate the direction of flow.

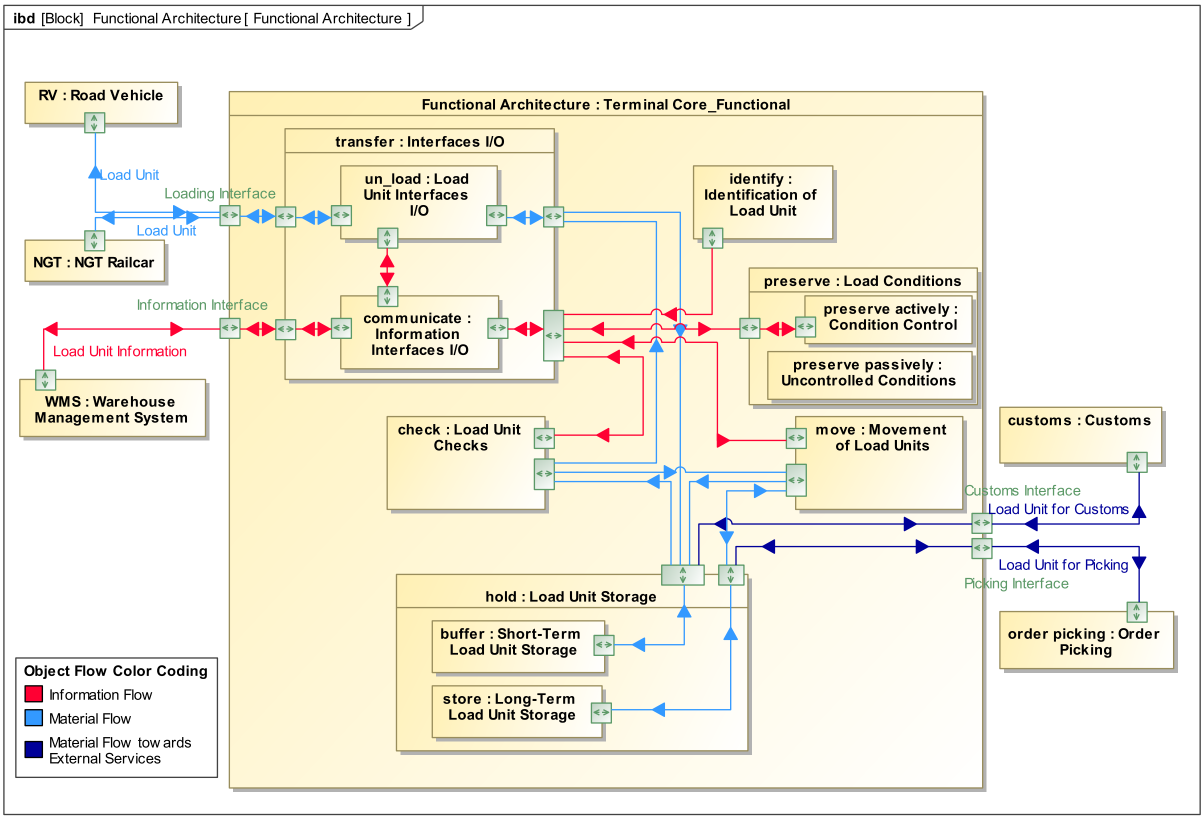


Figure 4-18: Functional architecture of the terminal core with interacting systems (SysML view)

The functional architecture developed by following the FAS methodology revealed that the key material handling functions are *un-/load*, *move*, *check*, and *hold* (*buffer* and *stor*) of the LU. Functional interfaces for material flow towards external services were implemented at the holding functions. The un-/loading, moving, and checking functions *communicate* with the WMS using information flows and the *information interface*. Further, *identifying* *of LU* and the *preserving of load condition* were identified as key functions of the terminal core. Different transshipment procedures (storage, crossdocking) were facilitated by the functional interfaces.

The functional architecture is the first comprehensive structural specification of the terminal core within this thesis. Yet, it has little in common with a typical physical system description, usually associated with structure. In contrast, the functional architecture is the most abstract representation of the system. The terminal core is reduced pragmatically to a level, where no technical solutions are stipulated. For the purpose of MBSE, which is to create a widely opened solution space in order to develop the most promising system variant (see Subsection 2.1.1), this abstract functional description is key. Based on this holistic functionality, the development of the logical architecture as first step towards a physical implementation was initiated.

4.5 Logical Architecture

Once the functional architecture was defined, the development of the logical architecture was done next. Both are structure-related descriptions of the SOI consisting out of several elements. In contrast to their functional counterparts, the elements of the **logical architecture** represent generic technical concepts and target the description of the SOI's logical operation (see modeling task 3 in Section 3.1).

In SYSMOD, the method for the development of the logical architecture proposes the use of sequence diagrams to identify the logical elements [Wei-2014, p. 148]. Although this was partly applied for the identification of top-level interactions with stakeholders before, the procedure was found not useful in this context. This was mainly caused by the characteristics of SysML, which prescribe the use of sequence diagrams for messages only[9]. Thus, no material flows can be displayed in sequence diagrams [Bra-2020]. Since the focus of this project was laid on the material flows, the added value by the usage of sequence diagrams was rather low in this context.

[9] Relict of the software-centered UML 2, which is the basis of SysML [Bra-2020].

Instead, it was decided to use the previously developed architectures as an initial basis for the derivation of logical elements. The base architecture (Figure 4-2) gave a basic idea of the necessary logical subsystems. However, it was regarded as being too general for a direct translation into a logical architecture. Thus, the functional architecture (Figure 4-18) was considered additionally. Both, functional and logical architecture describe the entire system comprehensively. Thus, every function had to be reflected by the entirety of logical elements.

Keeping the base architecture in mind, **logical elements** were developed based on functional elements as conceptual realizations of the corresponding functions. To underline the structural character, the logical elements were denoted as logical system blocks and implemented using the stereotype «logicalBlock». This stereotype was defined as a specification of the SysML «block» and was introduced to enhance discriminability of architecture blocks.

Figure 4-19 shows the mapping of logical system blocks to the functional elements using n:m allocation relationships. These allocations represent the key transformation from functions into logical blocks. This transformation is crucial to ensure a comprehensive realization of the abstract functional entirety by the logical architecture. Apart from the aspect

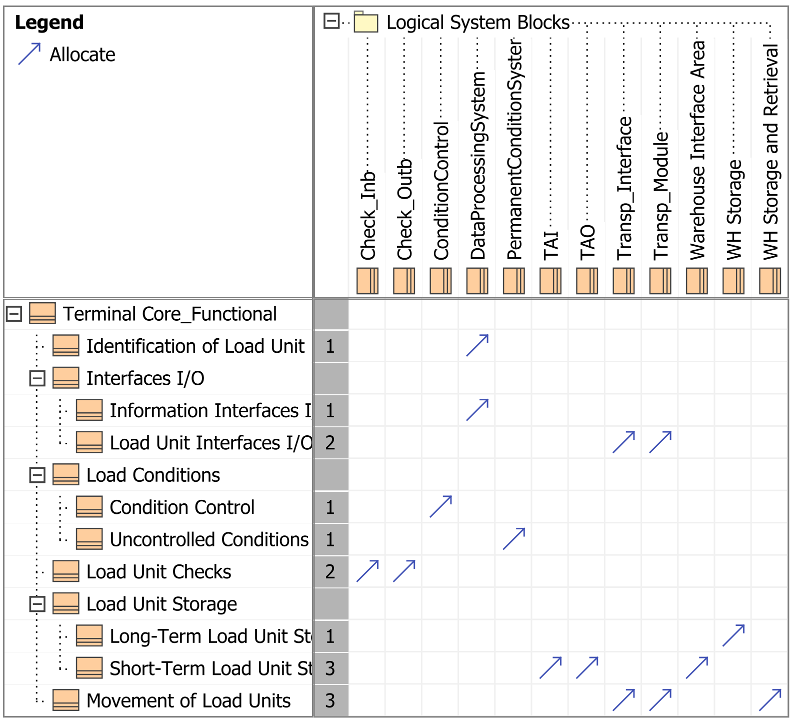


Figure 4-19: Mapping of logical elements to functional elements (SysML view)

of completeness, the allocations enabled a traceability from the system's behavior via its functional structure to its actual logical structure.

Corresponding to the storage block in the base architecture, initially a warehouse (WH) was created as logical subsystem. With ongoing progress it was subdivided into the *WIA*, a *WH storage and retrieval system (SRS)*, and the actual *WH storage*. These logical elements facilitate the logical operations of the store freight process (see Figure 4-9). Analogously, a transportation system was defined, consisting out of transportation modules (*Transp_Module*) and transportation interfaces (*Transp_Interface*). Due to similarities in logical operation of the functions *move* and *un-/load*, the logical elements of the transportation system facilitate both functions. Differences in checking operations led to the creation of separated logical elements for inbound checks (*check_inb*) and outbound checks (*check_outb*), both fulfilling the function *load unit checks*. To physically separate inbound and outbound flows, the differentiation of the logical system blocks *TAI* and *TAO* as main buffer areas was made. The functionality of *load conditions* was logically implemented defining a *Condition Control* and a *Permanent Condition System*. Finally, the *Data Processing System* was created as logical element to address both functions, *identify LU* and *communicate with WMS*.

Within an iterative process, the logical elements were gathered within **logical subsystems**. The logical subsystems served as intermediate abstraction level between the terminal core and the logical elements. The entirety of logical system blocks and corresponding logical subsystems is the **logical structure**. To get an idea about the correlation of functional and logical structure, Figure 4-20[10] shows the mapping of both structures as extended diagram view of Figure 4-19.

Similarly to the functional architecture, the logical architecture differs from its structure since it focuses on the interrelations and object flows among its elements. To realize these flows, the **logical interfaces** were modeled. As it was done within the functional procedure, proxy ports with flow properties were used to accomplish these interfaces with the different input and output flows. For each flowing item (*LU*, *LU Info*, *LU for customs* respectively *picking*) a type of port was created and denoted as **logical port**. Revealed by the modeling of logical

[10] Here, the allocation of logical elements to logical subsystems is anticipated, as it was mainly revealed later within the iterative specification of the logical architecture. Yet, the illustration was considered as helpful to provide a comprehensive understanding of functional and logical mapping.

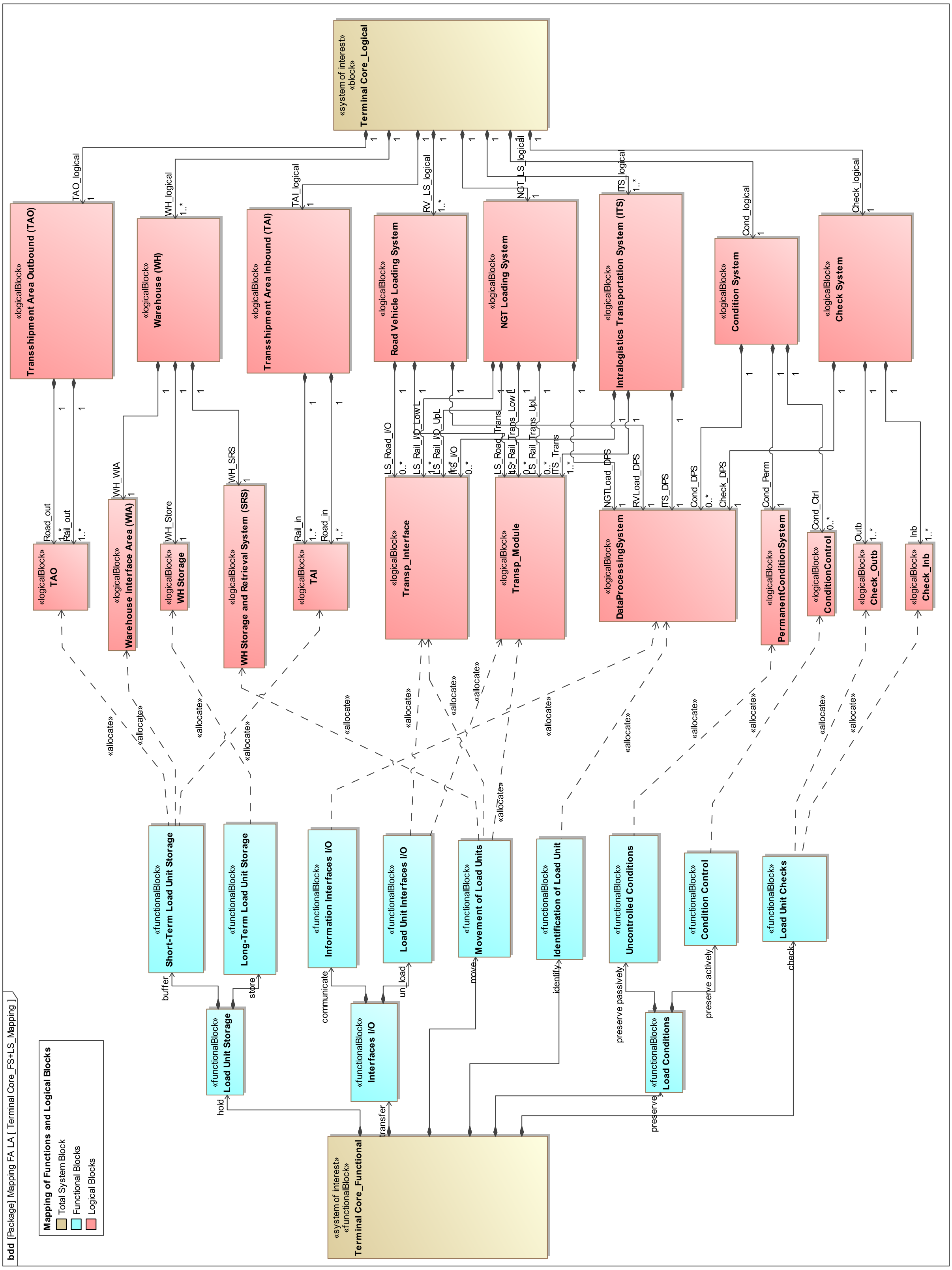


Figure 4-20: Mapping of logical structure to functional structure (SysML view)

elements, an additional logical port type was added for *controlled condition* flow. According to the representing functions and thus, the facilitating object flows, each logical element was provided with corresponding logical ports.

Apart from the logical elements, logical ports were assigned to the top-level system block (*terminal core*) as **logical system interfaces**. Since the logical architecture is classified as physically oriented concept, the logical system interfaces differ significantly from their functional counterparts. At this point, interfaces were no longer theoretical process junctions but discrete, physical interfaces. Driven by the focus of this thesis on internal material flow, this transition had only little impact on interfaces towards *WMS* (realized as single data interface) and *external services* (remained theoretical process interfaces[11]). In contrast, the logical system interface for the *LU* had to be differentiated according to the direction and type of mean of transport. For road vehicles, two spatial separated interfaces were defined (*GoodsReceipt*; *GoodsIssue*). Since the NGT CARGO railcar is a double deck vehicle, four different logical system interfaces were required to serve this mean of transport inbound respectively outbound and on the lower respectively upper level (*GoodsReceipt_LowL/UpL*; *GoodsIssue_LowL/UpL*).

Subsequently, the logical elements and subsystems were visualized in an internal block diagram of the terminal core. The visualized blocks were connected by object flows and rearranged according to process correlations. This whole iterative procedure was mainly driven by the findings from the system analysis (UC activities and system process). The process-centered rearrangements were key driver for final structuring of logical elements and subsystems, defining the logical structure (as anticipated by Figure 4-20).

The resulting **logical architecture** is presented by Figure 4-21. Again, the object flows are colored and the arrows indicate flow directions. For most of the material flows, the description of flowing item (*LU*, light blue) was hidden to enhance readability.

The basic logical subsystems are *loading system (NGT or road vehicle)*, *check system*, *TAI/TAO*, *intralogistics transportation system* (ITS), *condition system* and *WH*. Each subsystem consists out of several logical elements in specific roles. Logical elements of the

[11] According to preliminary considerations in Section 4.1.

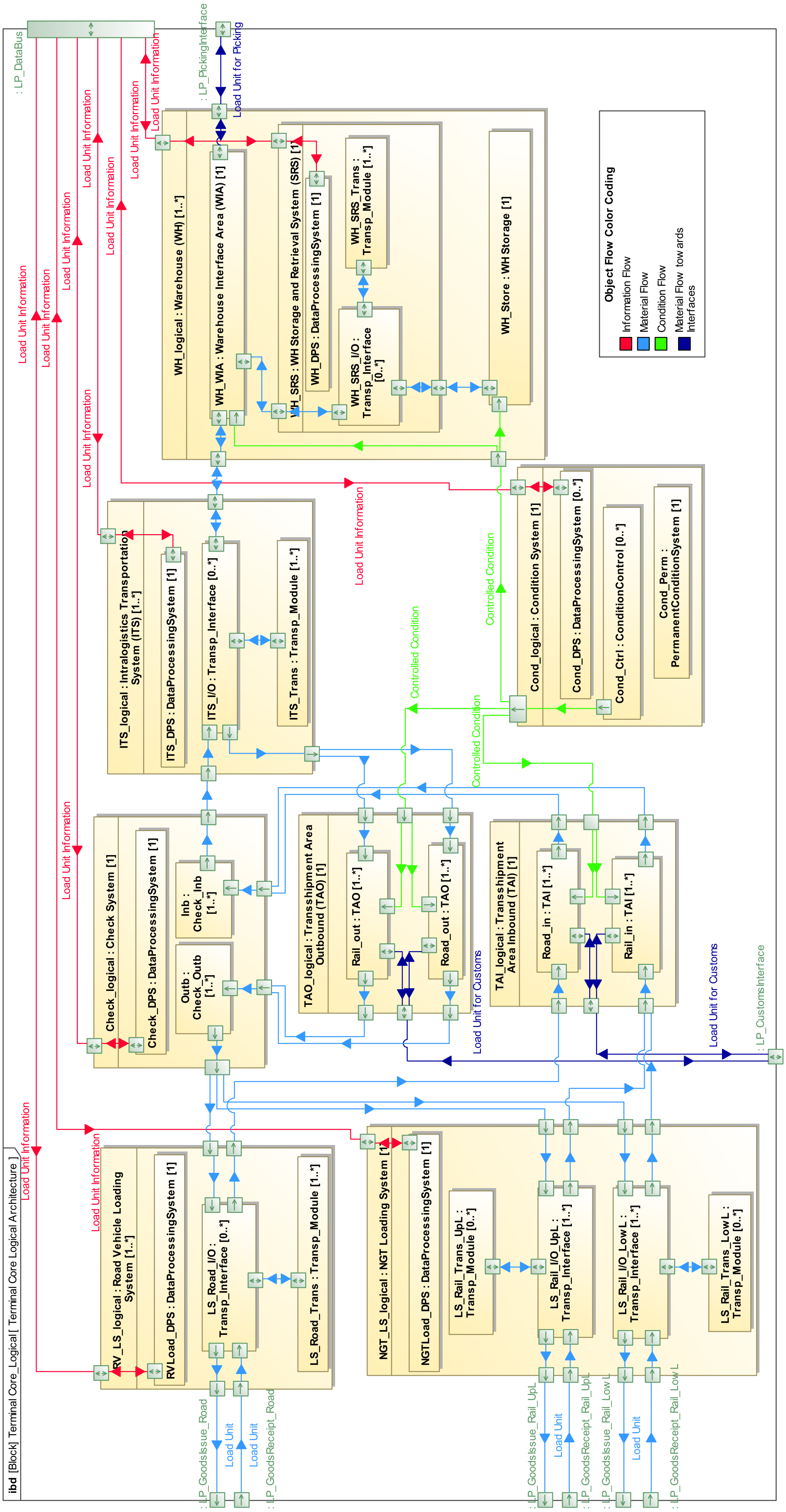


Figure 4-21: Logical architecture of the terminal core (SysML view)

same type occur in different roles within different subsystems. The figures in square brackets written in the header of each logical block refer to the possible multiplicity of the block. As an example, the terminal core was designed to contain at least one WH, although more than one WH are possible (denoted as [1..*]). The actual multiplicity depends on technical realization (technology, size) and is relevant for the design of a product architecture.

The logical architecture facilitates various transshipment directions and procedures. Firstly in the material flow, inbound LU pass the *loading system*. The *loading system* consists out of one or more *transportation modules*, and, depending on technical realization, one or more *transportation interfaces*. The logical elements were doubled in the *NGT loading system* for reasons of double deck loading. After being buffered in separated *TAI* (rail, road), inbound checks follow in the *check system*. In a next step, material flows were bundled and enter the *ITS*. Although the technical realization may differ, the logical structure of the *ITS* is comparable to the *loading system*. The *ITS* is the link from the inbound/outbound areas to the *WH*. The interface between *WH* and *ITS* is the *WIA*, where the LU are buffered and transferred to the *SRS* and subsequently put into *storage*. Since the *SRS* was specified as transportation system within the *WH*, the contained logical elements equal the *ITS*. The outbound procedure of the LU from storage to a vehicle basically equals the reverse inbound process. However, the outbound buffering takes place in the separated *TAO* (rail, road), followed by outbound checks before the LU is loaded. Crossdocking was realized by omitting the *WH* and directly transshipping the LU from the *ITS* towards the *TAO*.

Data communication and LU identification were realized as every logical subsystem except from holding areas was equipped with one *data processing system*. These elements were linked to the unique data bus as information interface to the WMS. The customs interface was implemented at the *TAO*/*TAI* and the picking interface was connected to the *WIA*. Finally, the *condition system* works independently from material flow. It was provided with a *permanent condition system* (referring to building systems), while a possible *condition control* and corresponding *data processing system* may be omitted depending on terminal design. Although it was out of core focus, key *controlled condition flows* were visualized.

Summarizing, the logical architecture was designed as intermediate step from the abstract functional description to a concrete physical implementation of structure. Apart from its interconnecting role, it is important to clarify the actual flows and sequences. Various design decisions were made, while keeping the logical architecture as generic as possible to enable a logical blueprint for physical variants. The logical subsystems imply initial physical units, which were used as basis for the development of product architecture variants.

4.6 Product Architecture

The product architecture is a specification of the logical architecture and equals the most detailed representation of the SOI within this system model. It consists out of concrete technical system components facilitating the object flow of interest. Being the lowest level of abstraction within the system model does not imply that the product architecture is the final step in the system development process. Much more, it is the baseline for the detailed specification of subsystems, or more specialized engineering models, such as CAD or simulation models [Wei-2016b, p. 62].

There is no unique solution for the product architecture. As stated in modeling task 4, the objective of this final modeling section was to derive **one exemplary variant of the product architecture**. Due to the increasing level of concreteness and complexity, the derivation of a product architecture was divided into several sections. As it was done for previous architectures, first the corresponding *product structure* was derived. Specifying the product structure and creating the solution space, a *logistics systems toolbox* was implemented. Subsequently, *evaluation criteria for intralogistics systems* were selected and tailored according to potential groups of goods. The modeling activities within this thesis were concluded by determining a *specific variant of a product architecture* based on the findings of the previous sections and facilitated by the logistics systems toolbox.

Product Structure

The initial baseline for the development of the product structure was the logical structure (Figure 4-20) and the allocation of functions to the logical elements (Figure 4-19). In consideration of these findings and additional literature research, the **product structure** was derived, consisting out of seven basic physical subsystems[12]. In detail, the physical subsystems involve *storage systems*, *conveyor systems*, *conveyor interfaces*, *freight check*

[12] To emphasize the real, technical character of the product blocks the term *physical* was used for elements of the product structure respectively architecture.

systems, *data servers*, *condition control systems*, and *building safety and security systems*. Subsequently, the product structure was mapped to the existing logical structure, to model the basic technical realization of each logical element (Figure 4-22).

With regard to the variety of solutions and the complexity of the terminal core, a particular multiplicity was allocated to each physical subsystem in the system model (as displayed in Figure 4-22). This enabled the modeling of combinations of different technical realizations for each physical subsystem and thus, a comprehensive physical description of the terminal core's product architecture.

Analogously to the logical blocks before, all physical blocks were modeled using the cus-

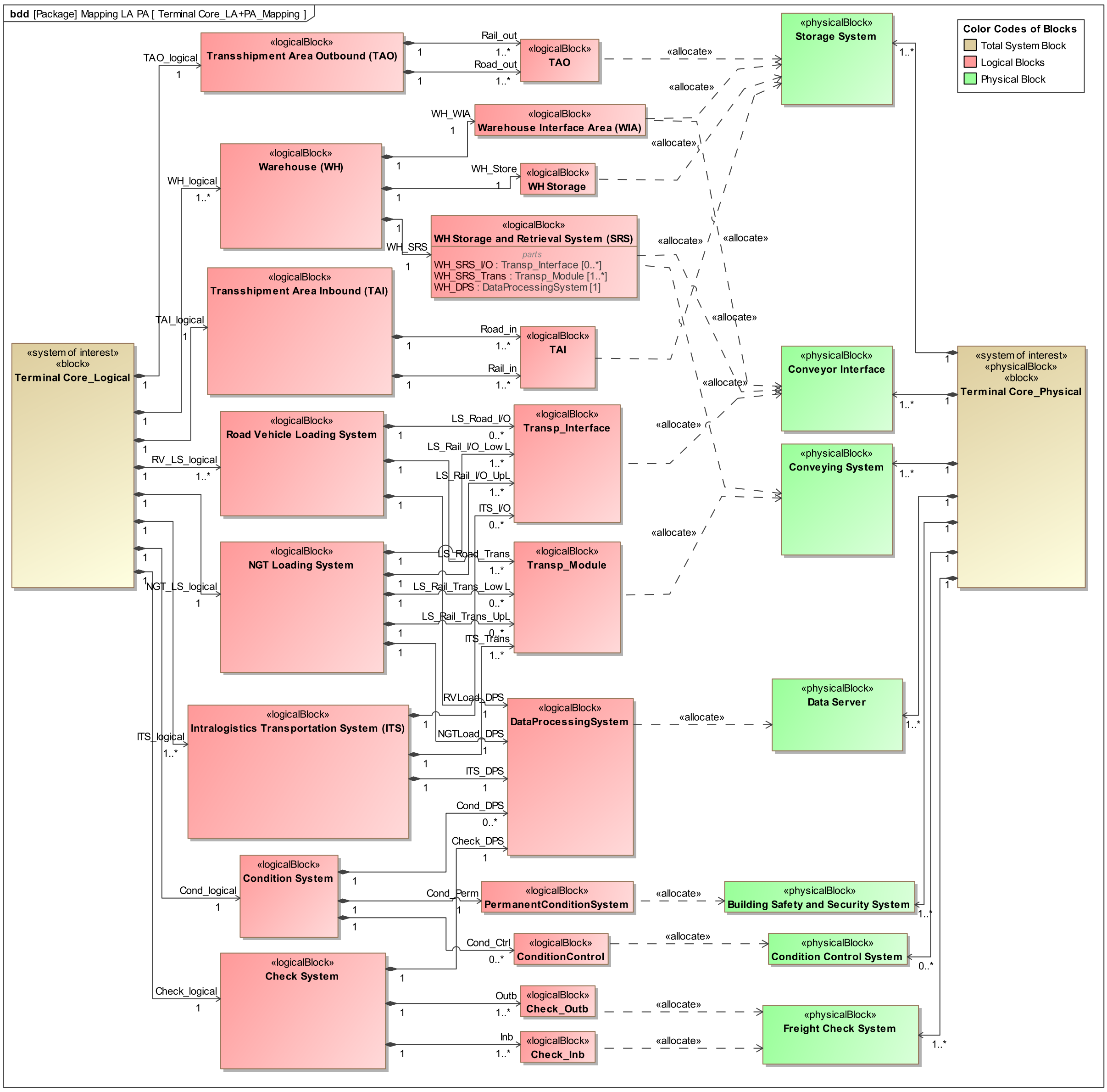


Figure 4-22: Mapping of product structure to logical structure (SysML view)

tomized stereotype «physicalBlock» as specification of the SysML «block» element. Further, ports as interfaces facilitating the flows among the elements had to be implemented. For reasons of scope, the considered flows in the product architecture were limited to the core material flow only. Therefore, **physical ports** were defined as interfaces facilitating the material flow. Since the physical blocks were regarded as the most concrete elements to specify, full ports with flow properties (*LU* as flowing item) were used to model the physical interfaces. According to the required material flows, physical ports were assigned to the physical subsystems defined in the product structure. On the SOI's top-level, the interfaces towards the vehicles (NGT, road vehicle) were modeled by assigning physical ports to the terminal core (analogously to the logical interfaces).

Logistics Systems Toolbox

The subsystems represent the basic physical parts required for a technical implementation of the terminal core. By definition, each subsystem was understood as a classification group of technical systems. Being rather generic, the physical subsystems had to be further specified to deliver concrete solution elements for a possible product architecture. Thus, a **logistics systems toolbox** was created. The logistics systems toolbox implemented concrete physical systems as lowest level of abstraction in the system model. The corresponding variants of all subsystems create the solution space of the terminal core.

Due to reasons of size it was not reasonable to display the whole solution space in an extended view of the product structure within this thesis. Instead, the toolbox is illustrated as independent sets of variants for each physical subsystem (Figure 4-23). For better orientation, the generic subsystems were highlighted. Every set of variants has to be understood as next-level specification of the physical subsystems displayed in Figure 4-22.

The focus of the toolbox was laid on the systems directly facilitating the material flow (*storage system*, *conveying interface*, *conveying system*, and *freight check system*). The corresponding variants within the toolbox were mainly selected based on *Hompel et al.* and *Schmidt* [Hom-2018; Sch-2019a]. Additionally, state-of-the-art industry solutions were considered [ATS-2021; GEB-2021]. Concerning the core intralogistics systems, additional theoretical information is given in Subsection 2.1.3. Regarding the focus of this thesis, the other subsystems in the toolbox were treated incidentally. Thus, the corresponding abstraction level was kept rather high. The variants listed in the toolbox were derived based

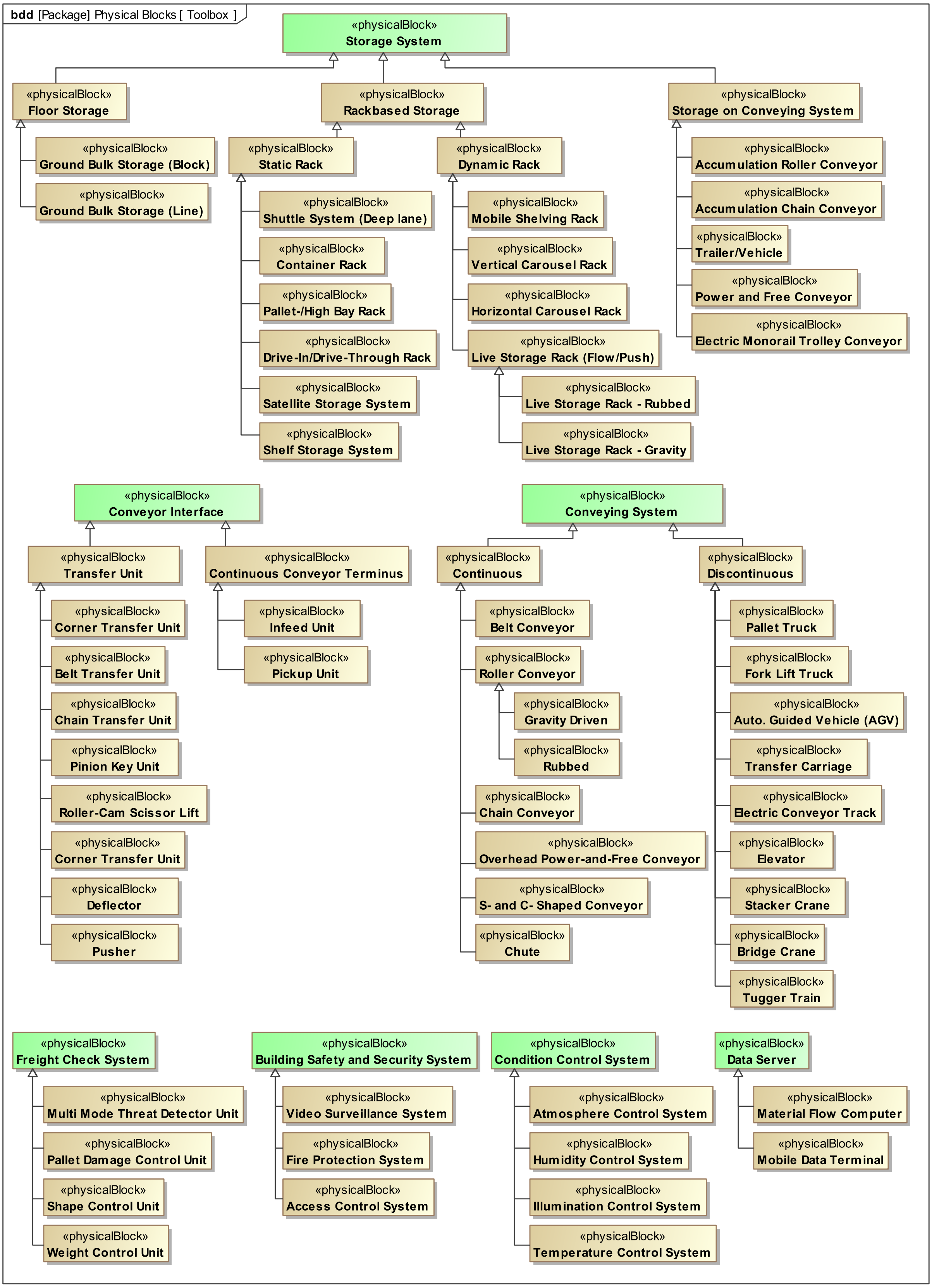


Figure 4-23: Implemented toolbox of technical variants; generic subsystems highlighted (SysML view)

on NGT CARGO project specifications (*building safety and security system* and *condition control system*) respectively based on *Günthner and Hompel* (*data server*) [Gün-2010].

In contrast to the previous structure developments (functional, logical, product), the specification of the toolbox did no longer refer to a logical or conceptual refinement of subsystems. Instead, each set of variants delivers various **equivalent technical solutions** for the corresponding subsystem. To realize these equivalences, SysML generalization relationships[13] were used to model the toolbox variants as displayed by Figure 4-23. On the one hand, this implies that every implemented variant is an equivalent and comprehensive representative of its superordinated subsystem. On the other hand, every variant has its own technical specifications and represents a different solution. Depending on the technical requirements of the design case, the most suitable variant can be selected for implementation.

It must be stated that the toolbox was not intended to be conclusive. Many more technical realizations of the subsystems exist which were not implemented in the system model. Yet, the toolbox provided a selection of common key variants facilitating intralogistics pallet handling and being sufficient within the scope of this thesis.

Tailoring of Evaluation Criteria for Intralogistics Systems

To enable adequate application of the toolbox in order to derive a feasible product architecture, a intralogistics system selection guideline was required. Therefore, **evaluation criteria** for the key material flow systems *conveying systems* and *storage systems* were determined based on literature.

Hompel et al. give a comprehensive overview over intralogistics systems and describe various criteria for the comparison and selection of these systems. For each criterion, a basic evaluation of the listed variants of key intralogistics systems is given (favorable, semi-favorable and unfavorable). Despite this, *Hompel et al.* remark that a tailoring and weighting according to the specific project is necessary [Hom-2018].

Not every intralogistics system mentioned by *Hompel et al.* was relevant for the NGT CARGO logistics terminal. Key driver for the selection of relevant system variants were the groups of goods, specified within previous work by DLR. These groups of goods were

[13] A SysML generalization relationship leads to an inheritance of properties from the parent to the child element (e.g. ports, values). Yet, the child element may exhibit additional, individual properties that differ from others.

divided into five clusters according to the criteria *type of load carrier*, *standardization of load carrier*, and *temperature control* (see Appendix A). Each intralogistics system listed by *Hompel et al.* was assessed in regard to these criteria. Thus, unsuitable variants could be excluded depending on the selection of a particular cluster of goods.

Analogously to the intralogistics system variants, not every evaluation criterion mentioned by *Hompel et al.* was relevant for the purpose of this thesis either. The overall application of the terminal core is the implementation in the NGT CARGO logistics terminal. *Woxenius* investigates on transshipment requirements of intermodal terminals in regard to the surrounding intermodal transportation network. For terminal types equivalent to the NGT CARGO logistics terminal, he identifies *capacity*, *rapid transshipment*, and *technical reliability* as key drivers for the terminal [Wox-2007]. Further, for the NGT CARGO logistics terminal a high, but realistic *degree of automation* is desired as major design requirement [Ehr-2020]. These key drivers guided the selection of relevant criteria.

For the conveying systems, six criteria were chosen from *Hompel et al.* [Hom-2018, pp. 240–242]. Five of these criteria were assigned to two clusters, namely *automation* and *technical complexity*, which were weighted 50% each. The sixth criterion referred to *storing capabilities* and was added without weighting impact to facilitate the evaluation of conveying systems for storage purposes. Selection criteria for speed and capacity were not given, as they are mainly influenced by the quantity of systems deployed [Hom-2018, p. 161]. The overall **evaluation matrix for conveying systems** is displayed in Figure D-1.

Similarly, seven criteria from the enumeration in literature were selected for the evaluation of storage systems [Hom-2018, pp. 120–121]. This time, four clusters were built and weighted 25% each. The clusters were *transshipment speed*, *capacity efficiency*, *technical reliability* and *automation*. The overall **evaluation matrix for storage systems** is displayed in Figure D-2.

Both evaluation matrices were applied to guide the selection of intralogistics systems when developing the product architecture. It must be mentioned that the matrices only represent a basic evaluation guideline. The overall number of parameters for selection of such systems is rather extensive and assumes detailed knowledge of the specific application case and the technical implementation. Further, general important criteria are weight and size of LU, desired throughput, number of articles, or investment cost [Hom-2018, p. 239]. However, in regard to the scope of this thesis, the basic evaluation possibilities given by the matrices were considered being sufficient.

Product Architecture Variant

As mentioned previously, the product architecture variant was designed for specific groups of goods. From the groups of goods introduced in Subsection 2.1.4 and clustered in Appendix A, the cluster 2 was selected, referring to **textiles** (group of goods 3) and **printings** (group of goods 4). This choice was met by the reason that both groups of goods in general are transshipped on standardized Euro pallets and do not require special transshipment conditions[14]. This decision had an impact on the requirements, as the requirements *cool chain* (2.1.1) and *storage humidity* (2.1.2) became irrelevant for the following product architecture. Further, both evaluation matrices for storage respectively conveyor systems were tailored to this cluster 2, as depicted in Appendix D.

Here, it is important to note that the consideration of another cluster of goods leads to different appropriate intralogistics systems and thus, to a different product architecture. Especially non-standardized load carriers or the need for precise transshipment conditions for perishable goods, such as vegetables, fruits, or pharmaceuticals, lead to numerous new requirements which have to be regarded [Ges-2021]. With an increase in number and detail of requirements, the required domain-knowledge about goods characteristics and logistics handling increases accordingly. To realize one possible product architecture within the limited time frame of this thesis, a transshipment of standardized Euro pallets without special condition requirements was regarded as being most promising.

The modeling of the chosen product architecture variant was facilitated by firstly introducing the concept of variants on the system model's top level. Figure 4-24 illustrates the basic implementation of a **top-level terminal core variant** in the system model.

To enhance the understanding of the model integration of the variant, the relation to the existing logical respectively physical system blocks is given within this view. The visualized system blocks represent the top-level nodes of the logical respectively product structure

[14] Simplifications were made, such as disregarding a possible hanging transshipment of clothes.

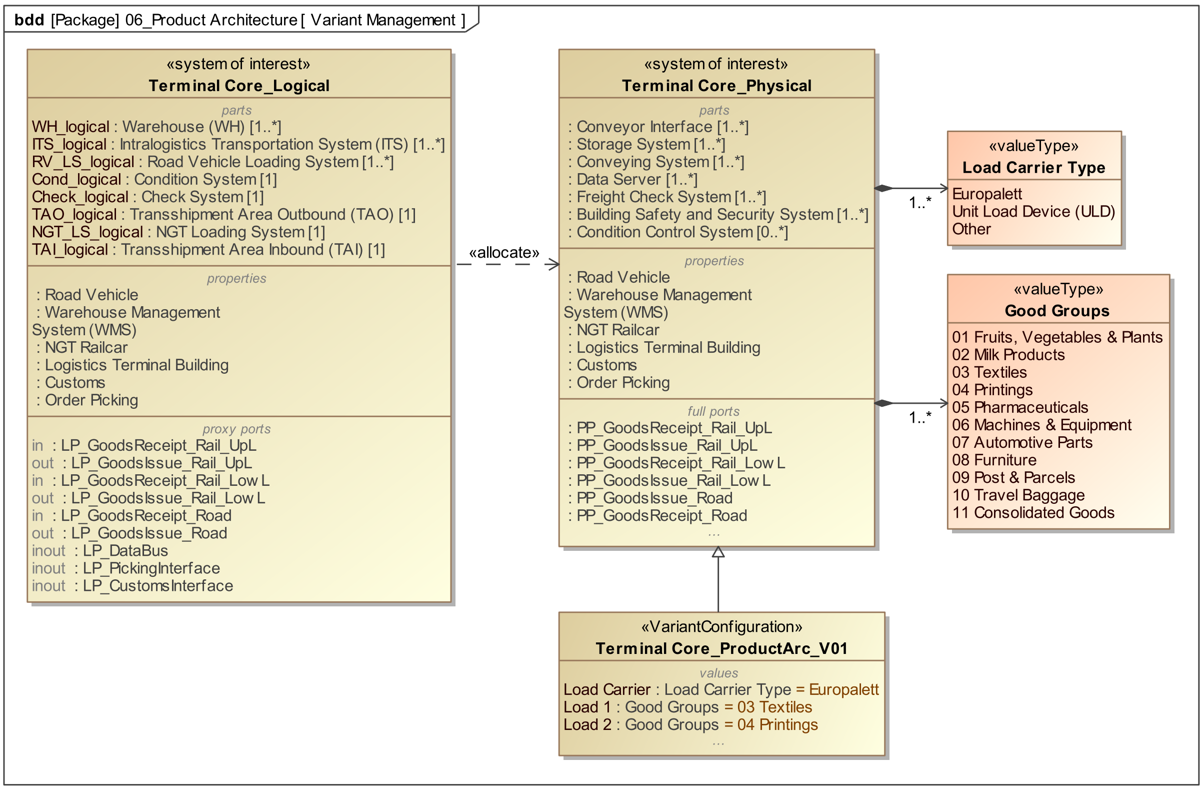


Figure 4-24: Model integration of terminal core variant V01 (SysML view)

and thus, a comprehensive structural description of the terminal core. Deduced from Figure 4-22, the logical terminal core block was allocated to its physical counterpart. The created variant of the physical terminal core system block was derived using the SysML generalization relationship (*TerminalCore_ProductArc_V01*).

Differentiation criteria for the terminal core's physical variants were implemented as *values*. The considered criteria were the *load carrier type* (Euro-pallet, ULD, other) and *groups of goods* (eleven groups respectively five clusters, see Appendix A). The values were associated to the physical terminal core block. An added multiplicity facilitated combinations of types of load carrier respectively groups of goods within one terminal core variant. Based on the previous decision on groups of goods, the groups *03 textiles* and *04 printings* and the type of load carrier *Euro pallet* were implemented as values of the *TerminalCore_ProductArc_V01* system block.

Once the terminal core variant was defined within the model, its product architecture was developed within an internal block diagram of the system block. The development of the product architecture was mainly facilitated by three findings of the previous modeling: the system's behavior prescribed the necessary processes, the logical architecture delivered the logical operations and corresponding structures, and the logistics systems toolbox finally provided the concrete technical systems for implementation. Figure 4-25 illustrates the **product architecture variant V01** as final model artifact of this thesis.

As stated previously when introducing the generic product structure, only core material flows were specified. In contrast to previous architectures, the color coding was changed. To provide orientation in flow directions, the colors represent different transshipment phases of the material flow (inbound, crossdocking, outbound). The spatial location of the external interfaces towards the vehicles was basically adopted from the logical architecture (road vehicle upper left corner, NGT lower left side). Caused by the increased level of detail and the resulting quantity of physical elements, it was only partly possible to adopt the arrangement of subsystems from the logical architecture and still provide a reasonable illustration.

All elements of the product architecture were taken from the toolbox (Figure 4-23) with the evaluation criteria (Appendix D) applied whenever possible. Supporting the selection process, individual properties as well as possible advantages or disadvantages of the systems were considered (introduced in Subsection 2.1.3). For model implementation, a unique role header was assigned to each element block to describe its role within the system and its allocation to the corresponding physical subsystem. Being out of scope, no quantities or size dimensions of the elements were taken into account.

In the following, the basic types of selected systems within the product architecture are presented, following a generic inbound - outbound procedure.

The loading systems (LS) were modeled differently for each mode of transport. As described in Section 1.2, the loading activities within the NGT CARGO railcar concept are executed using integrated systems. Together with an automated, exact positioning of the train cars in relation to the terminal core, the provision of fixed *roller conveyors* as docking interfaces was justified as appropriate. The total **LS for the NGT CARGO** was realized applying further *chain conveyors*, interconnected with *chain transfer units*. To cope with the discrepancy in height between the NGT CARGO upper level and the ground floor, an *elevator* module was selected from the toolbox for the inbound transshipment. Material flows

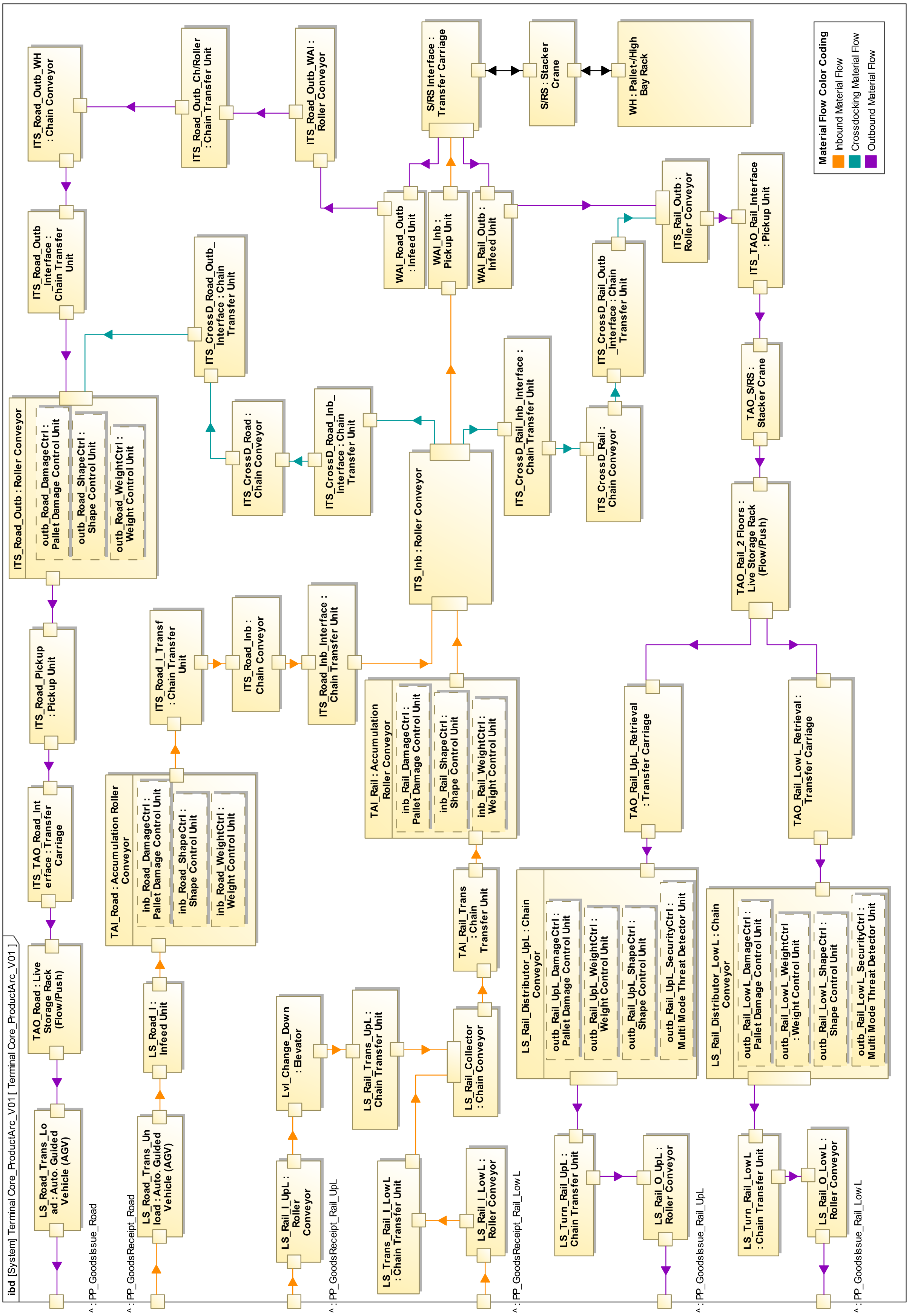


Figure 4-25: Product architecture variant V01 of the terminal core (SysML view)

from both levels of the NGT CARGO were merged on the lower level and passed over to the TAI (rail). The **LS for road vehicles** was implemented with different technology. In order to remain as independent as possible from the type of road vehicle, but still provide a high degree of automation, *automated guided vehicles* (AGV) were selected as LS for road vehicles. These AGV transship the inbound LU from the road vehicles to an *infeed unit* as interface towards the TAI (road). If the performance of the terminal core requires more than one infeed unit, a corresponding merging of the infeed flows has to be taken into account. Possible technical realization can be *transfer carriages* or *transfer chain units*.

Both **TAI**, road and rail, were implemented spatially separated as *accumulation roller conveyor*. The application of this type of holding facility was especially beneficial for a seamless and automated integration of buffer capacity within the material flow. Further, check systems are generally capable of being integrated in such roller conveyors. This was done for the **inbound checks** for road and rail. Inbound checks included a *pallet damage control*, a *shape control* and a *weight control* to facilitate all necessary inspections. Being excluded before, no physical facilities for the treatment of rejected LU or customs operations were implemented. Subsequent to the TAI, inbound flows from both modes of transport were merged within the ITS.

The core of the product architecture was the **ITS**, as it was designated to connect all other systems. The ITS was realized by implementing *roller conveyor* and *chain conveyor* elements, depending on the direction of the pallets[15]. These highly automated conveyors were selected as backbone of the ITS and were interconnected using mainly *chain transfer units* as conveyor interfaces. In addition to the result from the evaluation matrix, high throughput rates as well as low operating costs were decisive [Hom-2018, pp. 133–134]. The ITS was designed to facilitate the transshipment of LU from the TAI to the WH (inbound) and from the WH to the TAO (outbound). Further, crossdocking shortcuts were supplied by the ITS.

For the **WH**, a *pallet-/high bay rack* was implemented as core storage facility, operated by *stacker cranes* as SRS. A major point for the selection of the pallet rack was that it is easy to automate and it offers a high accessibility of each LU [Hom-2018, pp. 66–68]. This facilitates an efficient operation and independent storage cycles for the different stored goods,

[15] Due to their skid-like structure, Euro pallets (illustration given in Figure 2-11) must only be conveyed lengthwise on *roller conveyors* and crosswise on *chain conveyors* [Fei-2021].

which is advantageous for keeping the goods ready for different vehicles independently of each other. Due to lack of project data, decisive aspects for selection of storage systems such as storage duration or maximum quantity of stored goods could only be roughly approximated. Being a transshipment facility, the average freight dwell time in the terminal was evaluated as rather low and thus, high throughput rates were expected for the WH. From an economic perspective, the pallet rack was evaluated to be a reasonable choice due to its low throughput costs with increasing size together with reasonable storage space costs [Gud-2010, p. 565]. Being the interfacing area between the SRS and the ITS, the WIA was equipped with a *transfer carriage*, serving several *infeed* or *pickup units*. From here, the LU can be transshipped from the SRS to all directions within the ITS. Especially for the WIA and SRS it was important to deploy highly automated technology in order not to interrupt the automated process chain and consistent information flow already generated by the ITS.

In outbound direction, the ITS was designed to convey the LU from the WIA or its crossdocking branches to the TAO (road and rail). Again, both buffer facilities were spatially separated to prevent negative interference in performance [Hom-2018, p. 321]. Due to significant differences in design, the product architecture for the outbound branches is described in the following separately for rail and road.

Since the NGT CARGO concept stipulates the loading operations from the opposite train side compared to the unloading operations, the **TAO (rail)** was located on the opposite side of the rail tracks[16]. The corresponding crossing of the tracks was considered in the ITS rail outbound branch. For the TAO (rail) facility itself, a two leveled *live storage rack* was selected. As described in the theoretical section, this type of storage exhibits beneficial properties for its application as TAO, such as high throughput rates or its deterministic storage behavior. Storage operations were provided by a *stacker crane*, while retrieval operations were realized on the opposite side using a *transfer carriage* on each level. Subsequently, the outbound part of the **LS for the NGT CARGO** followed. In contrast to its inbound part, no vertical conveyor element was needed, since the allocation of the LU to a NGT CARGO level was managed by the TAO (rail) storage procedure. On the final path to the NGT CARGO, the **outbound checks (rail)** were integrated in the LS conveying system. Com-

[16] Rail tracks were not displayed in the product architecture.

pared to the inbound checks, a *multi mode threat detector unit* was added for reasons of security in high-speed train operations. The outbound checks were located at this very late point to keep the time between security check and loading as short as possible. Loading of the LU into the NGT CARGO was implemented analogously to the unloading procedure.

In contrast, no security checks were required for outbound LU heading to the road vehicles. As a consequence, the **outbound checks (road)** were located in prior to the TAO (road) and thus, integrated into an ITS *roller conveyor*. This offered the benefit of providing more time for failed check management. The check design was equivalent to the inbound checks and covered *control units for weight*, *shape*, and *pallet damages*. The subsequent TAO (road) was again implemented as *live storage rack*. Analogously to the inbound procedure, **LS for road vehicles** facilitated the transshipment of LU from the TAO (road) to the vehicles. For equivalent reasons, *AGV* systems were implemented for this task.

Compared to the product structure in Figure 4-22, certain physical subsystems were not visualized in the view of the product architecture variant (Figure 4-25).

The generic physical subsystem *data server* was intended to realize the information-related functions. The material flow systems applied so far in the product architecture variant mostly rely on deterministic continuous conveyors and exhibit a high degree of automation. Thus, a central *material flow computer* was preferred over *mobile data terminals* to handle the information management and control efficiently. Due to the focus on the material flow only, the material flow computer is not displayed the product architecture view.

Further, a *building safety and security system* was considered to meet the required functionality of permanent conditions. Here, a strong relation to the terminal building and its systems is given, which was not in the focus of this thesis. Analogously to the material flow computer, the product architecture view does not show this system either.

Apart from that, as the requirements of the chosen groups of goods do not demand controlled conditions, a *condition control system* was not implemented in this variant at all.

Concluding, the developed product architecture variant is one possible technical implementation to realize the transshipping material flow of the goods textiles and printings. As the previously introduced architectures, the product architecture is an independent and comprehensive description of the terminal core's structure. Furthermore, it is the most concrete specification targeted by this thesis. In this role, the product architecture is the key artifact in the following verification and validation process.

5 Verification of the Terminal System Architecture

In the research approach guiding this thesis, the developed system architecture equals the actual support as outcome of the PS phase (see Section 1.4). Concluding the PS, a verification is recommended for the purpose of *support evaluation* [Ble-2009, p. 176]. Applied to MBSE context, a verification targets evidence that the developed system architecture is implemented correctly [Mad-2018]. The correctness of a system's architecture implies a formal aspect (Section 5.1) as well as a content-related aspect (Section 5.2).

5.1 Formal Verification

The first aspect of correctness refers to the application of SysML as modeling language facilitating the modeling. In regard to the modeling language, correctness indicates that the system model is *free of syntactic and semantic errors* [Mad-2018].

Basic support addressing the formal aspect of correctness was provided by the modeling tool Cameo Systems Modeler as it constantly checks consistency of model inputs and creates warnings in terms of misuse of syntax [NoM-2021]. However, there is no way to guarantee error-free implementations of all model artifacts. Hence, a comprehensive, mathematical verification of a model of this size is not possible [Mad-2018; Bra-2020].

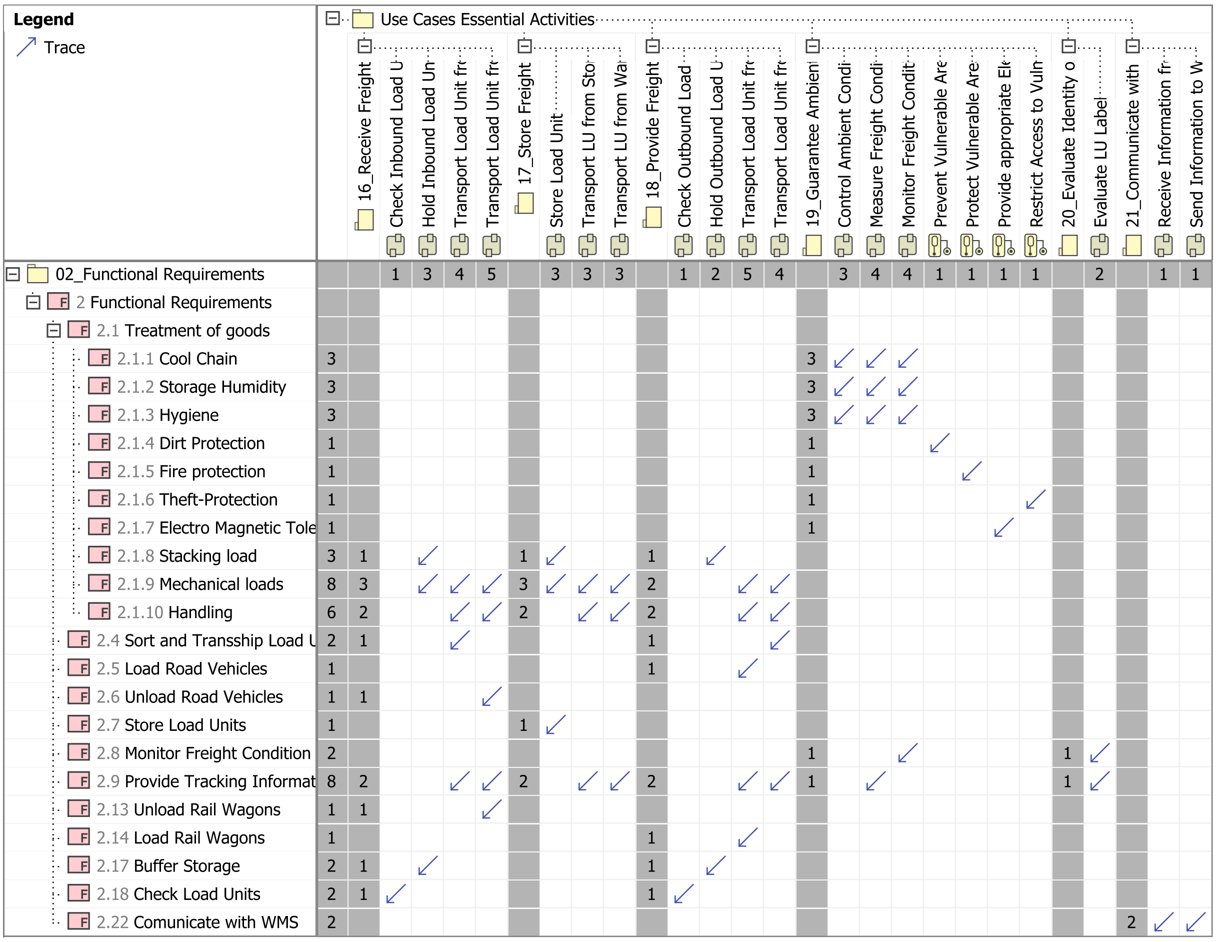
A more practicable possibility of verification are peer reviews. Thus, an interview with a SysML expert was conducted in order to receive basic feedback on the correct formal application of SysML [Bra-2021]. Due to the complexity of the model, a comprehensive examination of every view could not be made in the short time available. Despite this, by processing the expert's feedback a sufficient confirmation of the formal correctness of the model was achieved.

5.2 Content-Related Verification

The second aspect of correctness refers to the application of the methodology and thus, to the related modeling content. Here, the verification's task is to demonstrate that the methodology was applied correctly so that the model is *complete*, *consistent* and the artifacts are *traceable* [Mad-2018]. *Blessing and Chakrabarti* add, that the model's verification should ensure that *'[...] the detailed functionality [of the model] has a strong chance of realizing the intended impact'* [Ble-2009, p. 177]. Applied on the developed model of the terminal core, this means that evidence for the traceable fulfillment of the functional requirements by the product architecture has to be given.

The most detailed level of functionality within the model is the process description in the UC activities (see Subsection 4.3.4). To ensure the functional completeness of the model, tracing relationships among the essential activities and the functional requirements were implemented (Figure 5-1). As illustrated in this view, every functional requirement of the terminal core was addressed by at least one essential activity.

A consistent and complete transfer of the requirement tracing from the terminal core's behavior to its structure was ensured by the stringent application of the FAS methodology.



	Use Cases Essential Activities	16_Receive Freight	Check Inbound Load U	Hold Inbound Load Un	Transport Load Unit fr	Transport Load Unit fr	17_Store Freight	Store Load Unit	Transport LU from Sto	Transport LU from Wa	18_Provide Freight	Check Outbound Load	Hold Outbound Load U	Transport Load Unit fr	Transport Load Unit fr	19_Guarantee Ambient	Control Ambient Condi	Measure Freight Condi	Monitor Freight Condit	Prevent Vulnerable Are	Protect Vulnerable Are	Provide appropriate El	Restrict Access to Vuln	20_Evaluate Identity o	Evaluate LU Label	21_Communicate with	Receive Information fr	Send Information to W
02_Functional Requirements			1	3	4	5		3	3	3		1	2	5	4		3	4	4	1	1	1	1		2		1	1
2 Functional Requirements																												
2.1 Treatment of goods																												
2.1.1 Cool Chain	3															3	↙	↙	↙									
2.1.2 Storage Humidity	3															3	↙	↙	↙									
2.1.3 Hygiene	3															3	↙	↙	↙									
2.1.4 Dirt Protection	1															1				↙								
2.1.5 Fire protection	1															1					↙							
2.1.6 Theft-Protection	1															1							↙					
2.1.7 Electro Magnetic Tole	1															1						↙						
2.1.8 Stacking load	3	1		↙			1	↙			1		↙															
2.1.9 Mechanical loads	8	3		↙	↙	↙	3	↙	↙	↙	2			↙	↙													
2.1.10 Handling	6	2			↙	↙	2		↙	↙	2			↙	↙													
2.4 Sort and Transship Load U	2	1			↙						1				↙													
2.5 Load Road Vehicles	1										1			↙														
2.6 Unload Road Vehicles	1	1				↙																						
2.7 Store Load Units	1						1	↙																				
2.8 Monitor Freight Condition	2															1			↙					1	↙			
2.9 Provide Tracking Informat	8	2			↙	↙	2		↙	↙	2			↙	↙	1		↙						1	↙			
2.13 Unload Rail Wagons	1	1				↙																						
2.14 Load Rail Wagons	1										1			↙														
2.17 Buffer Storage	2	1		↙							1		↙															
2.18 Check Load Units	2	1	↙								1	↙																
2.22 Comunicate with WMS	2																									2	↙	↙

Figure 5-1: Tracing of essential activities (system behavior) to functional requirements (SysML view)

Following the modeling approach, the essential activities were traced to functional groups (Figure 4-16), which in turn were mapped to the functional elements of the functional structure (Figure 4-17). Figure 5-2 shows the resulting tracing of the functional requirements to the functional elements as first structural level of the terminal core.

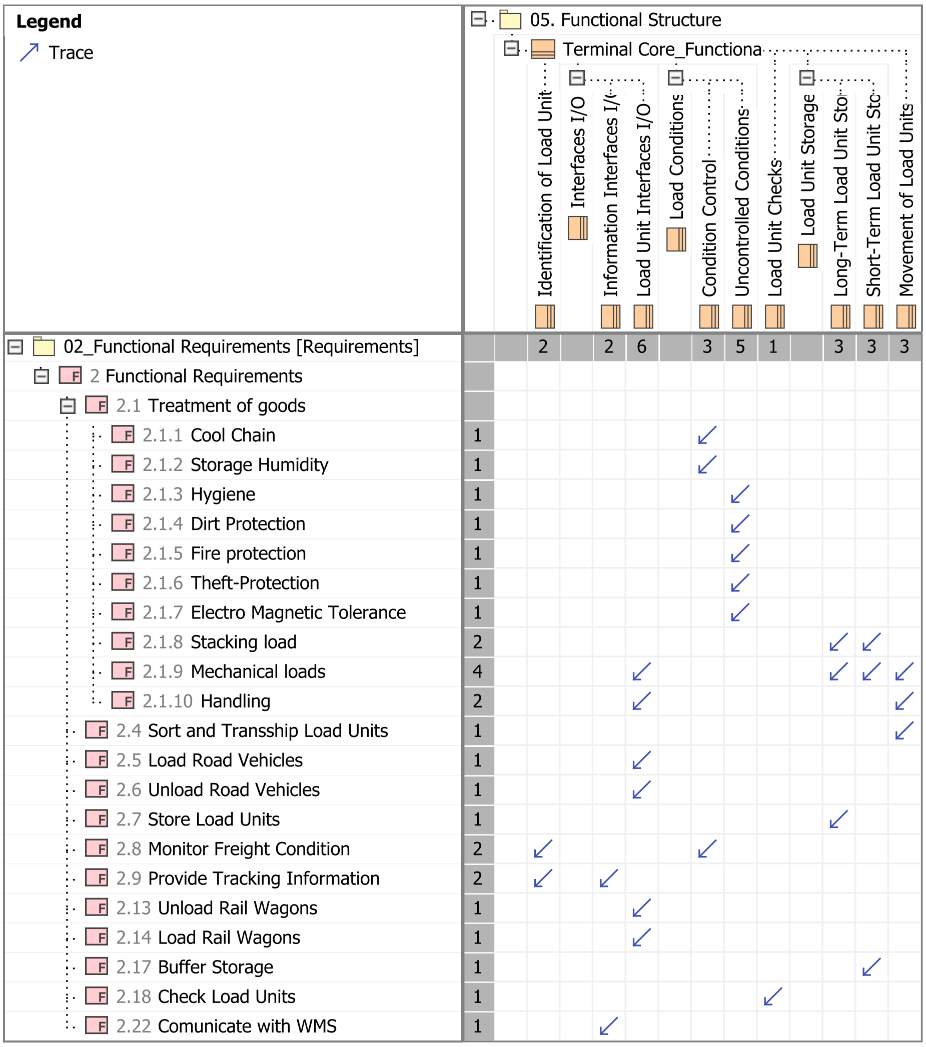


Figure 5-2: Tracing of functional elements (system structure) to functional requirements (SysML view)

In the further modeling, a consequent allocation of elements for each structure to its preceding structure was done to preserve the compliance of the architecture to the requirements. These allocations were already partly shown by Figure 4-20 and Figure 4-22. Final artifact of this allocation procedure is the product architecture for the terminal core variant V01. To illustrate the comprehensive structural allocations, Figure 5-3 displays the mapping of all elements of the terminal core variant V01, including functional structure, logical structure, product structure, and physical implementation. This figure demonstrates a complete, consistent and traceable picture of the SOI's structure implemented in the model.

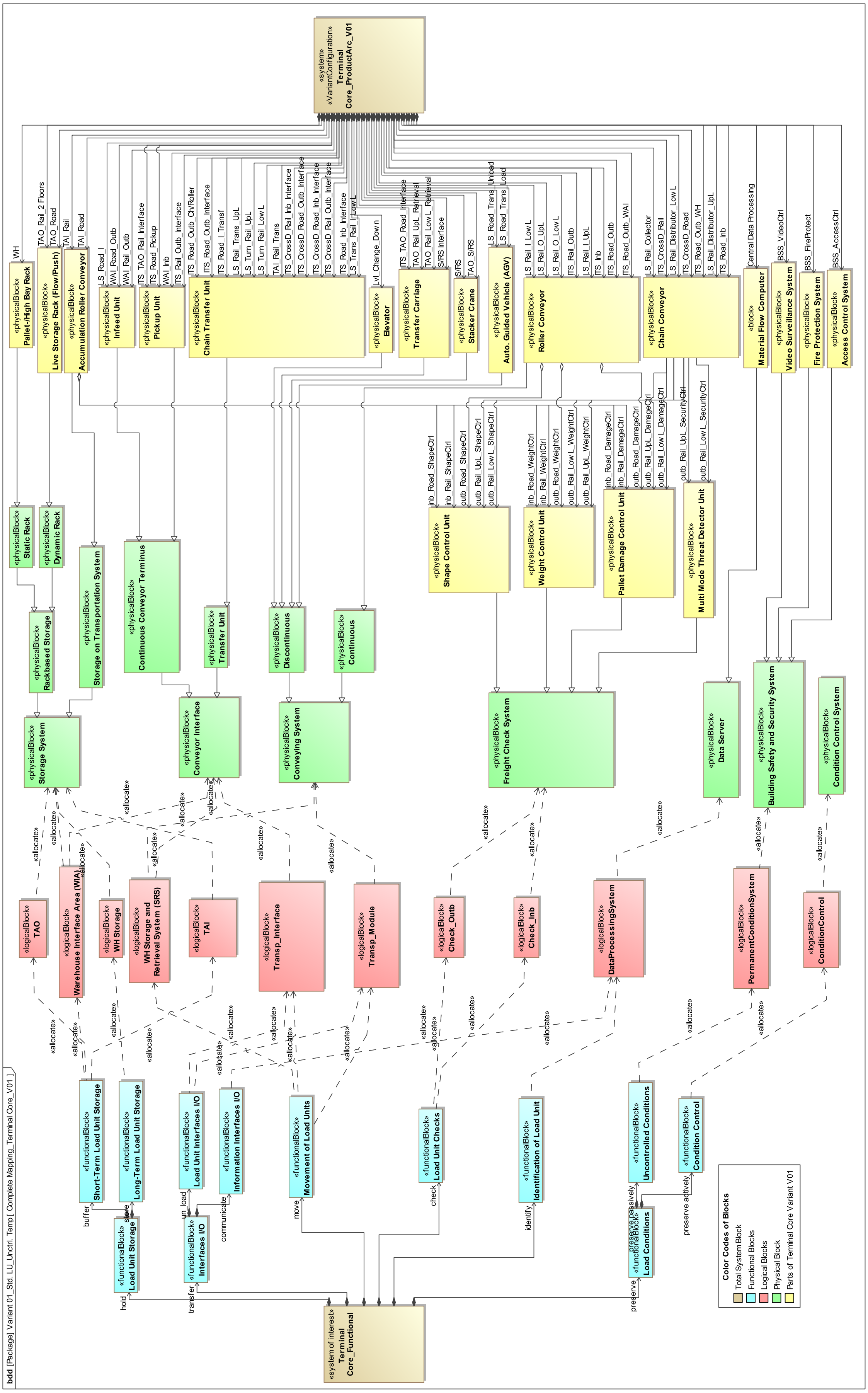


Figure 5-3: Comprehensive mapping of structure of the terminal core variant V01 (SysML view)

As final part of the verification, the consistency of total tracing shall be demonstrated. For reasons of size and readability, it was not practical to illustrate the tracing of all the requirements to all physical elements applied in the product architecture of the terminal core variant V01 within one view. However, to give an idea about the implemented traceability, Figure 5-4 displays the consistent tracing path between the physical element *LS_Rail_I_LowL* and the requirement *unload rail wagons*. It shows the modeled relationships from the specification via the behavior to the structure, down to the individual physical element.

All in all, this was considered as sufficient evidence for the content-related correctness of the model. Thus, the verification was successful.

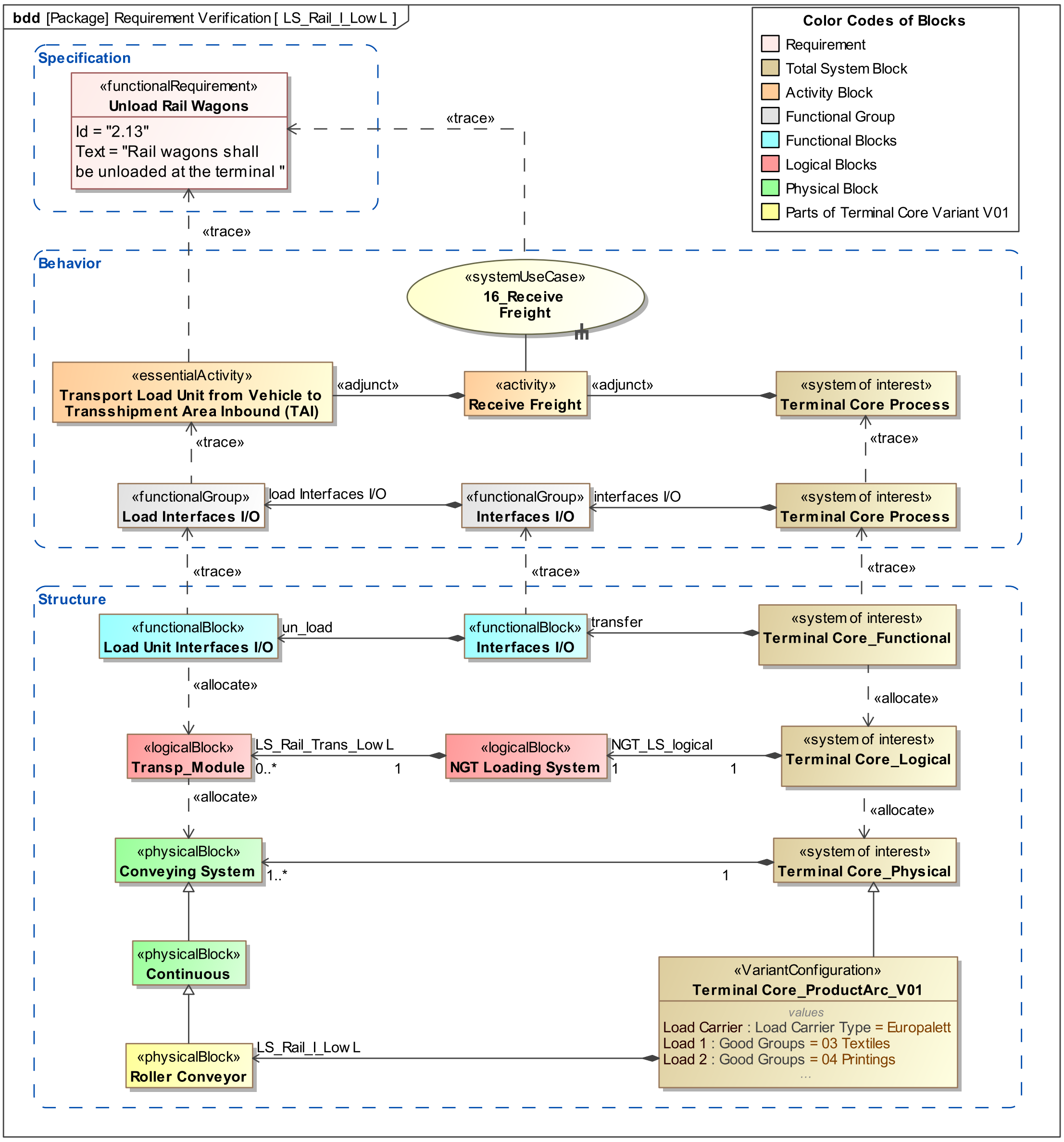


Figure 5-4: Tracing path from physical element to functional requirement, using the element 'LS_Rail_I_LowL' and the requirement 'unload rail wagons' as examples (SysML view)

6 Evaluation of the Terminal System Architecture

In the previous chapters, the performed comprehensive PS phase delivered a verified terminal system architecture as actual support of the research approach. The following chapter covers an evaluation of the terminal system architecture as part of an initial DS-II phase [Ble-2009, p. 195]. The evaluation targets a discussion about the applicability of the terminal system architecture in regard to its intended use. Further the findings in concerning the overall research project are discussed.

The evaluation starts with a validation to demonstrate the terminal system architecture's ability to meet the desired requirements and its general plausibility (Section 6.1). Subsequently, the fulfillment of the research objectives is evaluated in Section 6.2 to answer the research question. Finally, the overall research project is discussed (Section 6.3).

6.1 Validation of the Terminal System Architecture

The previously executed verification gives evidence that the terminal system architecture was formally built correct and theoretically fulfills the requirements. In contrast, a validation is generally conducted to prove that a system achieves its *intended use* in its intended operational environment [ISO-15288]. This implies investigation on the *suitability of initial assumptions* as well as on the system's ability to *comply to the intended requirements from a domain-specific perspective* [Fri-2014, p. 22]. Applied to this thesis, this means that the terminal system architecture must be reasonable from a logistics point of view, taking into account the assumptions. Further, the developed solution has to be evaluated whether the desired requirements are actually met.

For the validation, the consultation of external domain-specific experts is recommended for two reasons. Firstly, the expert has the required domain knowledge to judge about the fulfillment of requirements. Secondly, an external person does not exhibit a familiarization bias to the system aimed at by the validation [Fri-2014, p. 22; Ble-2009, p. 177].

A first interview with a logistics expert was conducted at the beginning of the modeling process, after the initial intralogistics process (Figure 4-3) was defined [Fei-2020]. This interview aimed at the validation of the model input and assumptions, such as the focus on material flow respectively on pallets as load carriers, or the exclusion of order picking and customs. Further, the basic understanding of sub-processes within the intralogistics process was confirmed. Subsequent to the interview, the model was adjusted according to the lessons learned from the interview.

The validation of the final system was the objective of a second logistics expert interview [Fei-2021]. Once more, the system's complexity and the short time available did not allow to validate the total architecture. Thus, only the product architecture of the terminal core variant V01 as key artifact of this thesis was validated. However, the SysML view of the product architecture (see Figure 4-25) is difficult to read and to understand for persons without SysML background. To cope this issue, the product architecture was pragmatically visualized using simple graphical elements in Microsoft PowerPoint. This visualization was the baseline in the expert interview. Figure 6-1 shows an updated version of this simplified

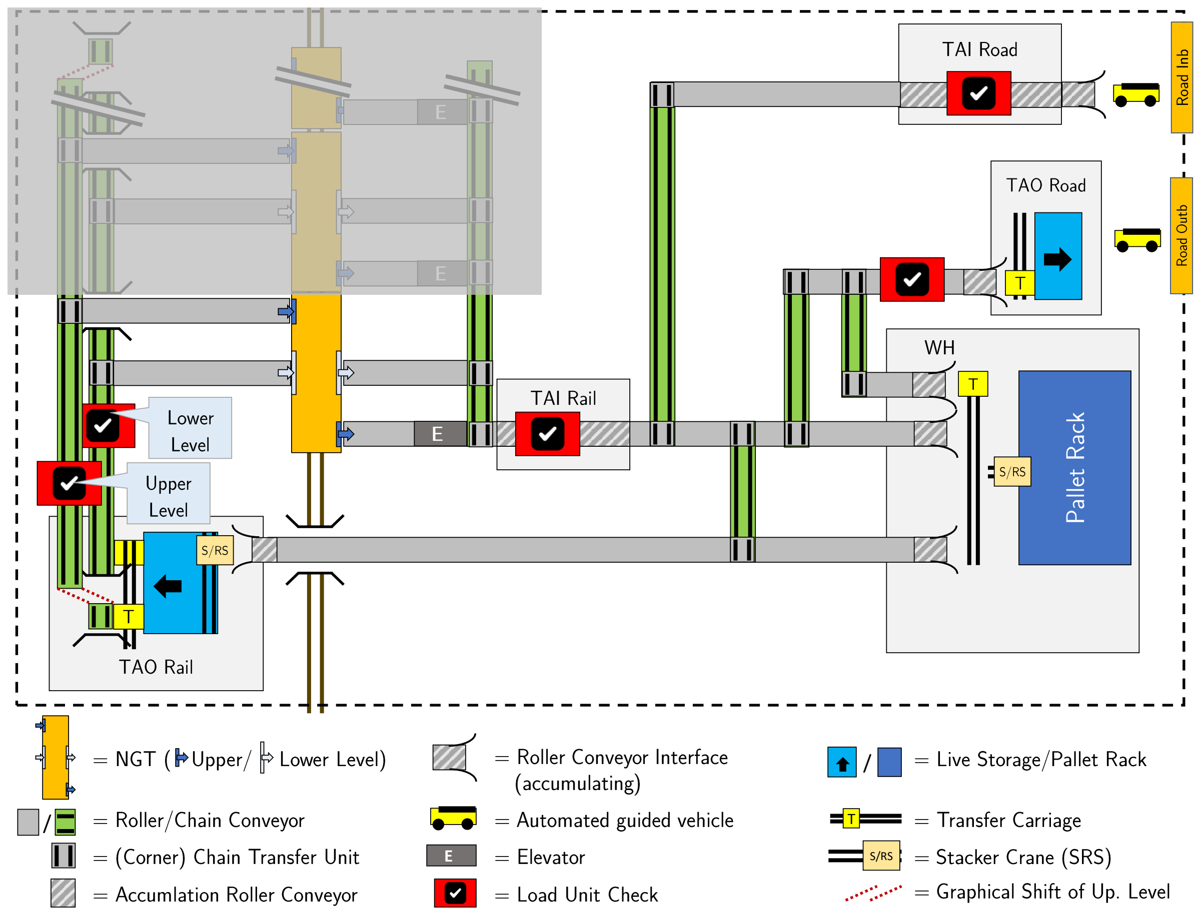


Figure 6-1: Simplified graphical visualization of the product architecture variant V01; multiple NGT railcars are grayed out since they are not part of Figure 4-25 (own illustration)

visualization. It already considers the feedback received in the expert interview, such as pallet heading direction, aspects of track crossing, or end-to-end automation. As a general evaluation result, the product architecture was considered a basically feasible concept from a logistics perspective.

Apart from general feedback on applied logistics concepts, key outcome of the second expert interview was an evaluation of the product architecture concerning its compliance to the requirements. Therefore, the requirements listened in Appendix C were discussed and evaluated in regard of their degree of fulfillment. This resulted in four clusters of requirements, namely *fulfilled*, *partially fulfilled*, *failed*, and *irrelevant*.

Figure 6-2 shows the quantities of assigned requirements for each cluster. From the total 31 requirements, 19 were directly or indirectly fulfilled by the developed system. Yet, the 12 remaining requirements were not 100% met. 2/12 requirements were regarded as irrelevant while no compliance of the system was achieved for 3/12 requirements. The remaining 7/12 requirements were only partially fulfilled.

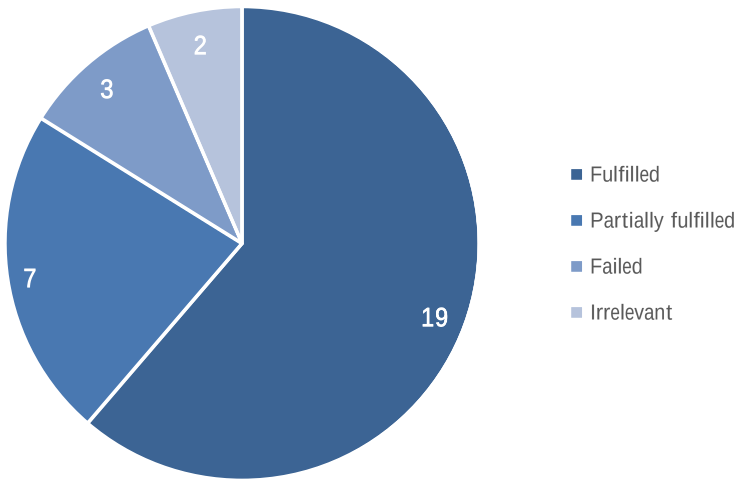


Figure 6-2: Fulfillment of requirements based on expert interview (own illustration)

As already touched on in Section 4.6, the requirements *cool chain* (2.1.1) and *storage humidity* (2.1.2) were regarded being *irrelevant*. Reason for this is that the selected groups of goods to handle (textiles and printings) were supposed not to require controlled conditions. In case of selecting a different cluster of goods to transship, these requirements may become relevant.

The modeled terminal core could not be evaluated conclusively in regard to the performance requirements *NGT railcar turnaround time* (5.1), *energy consumption* (5.4), and *degree of capacity utilization* (5.5). Reasons for this are missing level of detail, the static character of the model, and the lack of modeled parametrics. Thus, all of these requirements were denoted as *failed*, since the model was not capable to give evidence for fulfillment[1].

However, a rough assessment of the performance requirements was executed in the expert interview. The terminal system architecture was estimated as likely to be able to fulfill the requirement 5.4 by applying general energy efficient technology or stacker cranes and AGV with energy recovery. Due to the high degree of automation, the terminal may exhibit energy saving potential in regard to illumination, which was claimed to be one of the key drivers of the terminal's energy consumption. In contrast the requirements 5.1 and 5.5 were estimated as more likely not to be satisfied by the terminal architecture. An evaluation of the loading performance for the NGT railcars based on the static concept according to Figure 6-1 is barely possible. Yet, considering a capacity up to 1000 pallets per NGT CARGO block train[2], the architecture design was hardly expected to facilitate a *five minute turnaround time* for an NGT CARGO (as stipulated by requirement 5.1). Further, train operation characteristics result in a highly volatile volume of LU to transship. Relying mainly on continuous conveyors, the terminal (especially LS and ITS) is designed according to the peak volume and high throughput of LU. As a downside of this design decision, the terminal exhibits overdesign in off-peak operation phases. Consequently, a reasonable *degree of capacity utilization* (5.5) was estimated as hard to achieve for the developed layout. However, a conclusive evaluation on the performance requirements needs further investigation.

The rating of *partial fulfillment* of requirements was given for different reasons. The requirement *load unit size* (4.2) stipulates transshipment of pallets and ULD. Yet, the modeled terminal is only capable of transshipping pallets. The transshipment of ULD requires different systems, which were not implemented. Hence, this requirement was evaluated as partially fulfilled. The following six requirements were denoted as partially fulfilled, too:

- *Monitor freight condition* (2.8)
- *Provide tracking information* (2.9)

[1] It may be regarded that *Friedenthal et al.* claim the issue of failed validation of performance requirements to be inherent to modeling languages representing only process and/or functional flows [Fri-2014, p. 23].
[2] According to the NGT CARGO specification by DLR [Böh-2017].

- *Communicate with WMS* (2.22)
- *Order picking interface* (4.3)
- *Customs interface* (4.4)
- *Damaged load* (6.1)

All these requirements share the consideration of aspects, which were not excluded from the modeling per se, but were not realized in the product architecture either. This mainly concerns the missing physical implementation of the information flow. For example, a material flow computer and a video surveillance system were implemented in the terminal variant's basic product structure to address requirements 2.8, 2.9 and 2.22 (see yellow blocks in Figure 5-3). Since they do not physically facilitate the material flow, they were not integrated into the developed product architecture. In addition, the interfaces towards picking and customs (4.3, 4.4) and the treatment of rejected LU (6.1) were covered on a process level (see Figure 4-7), but they were not considered in the product architecture either. Despite this, as a basic compliance to these requirements is given by the terminal on a functional and logical level, they were claimed to be partially fulfilled.

The remaining requirements listed in Appendix C were considered as being satisfied by the terminal system architecture. Especially in regard to general material flow and the degree of automation the concept was evaluated being beneficial. A consequence mentioned was that pallet damage checks may be reduced to inbound (road) only, as a fully automated handling system usually does not harm pallets. Apart from a safe and reliable transshipment, a consistent chain of automated handling simplifies tracking of LU, as the position of each LU is deterministic and known to the material flow computer. This was denoted as major capability of state-of-the-art transshipment terminals, which is the intended use of the developed terminal system architecture. Consequently, the system was evaluated as basically achieving its intended use in its intended environment.

Based on the lessons learned from the second expert interview, a digital rendering of the terminal was developed in collaboration with DLR. Reasons for a more detailed graphical implementation were to further testify the plausibility of the developed concept and to create comprehensible communication media. Figure 6-3 shows this visualization of the terminal core variant developed within this thesis.

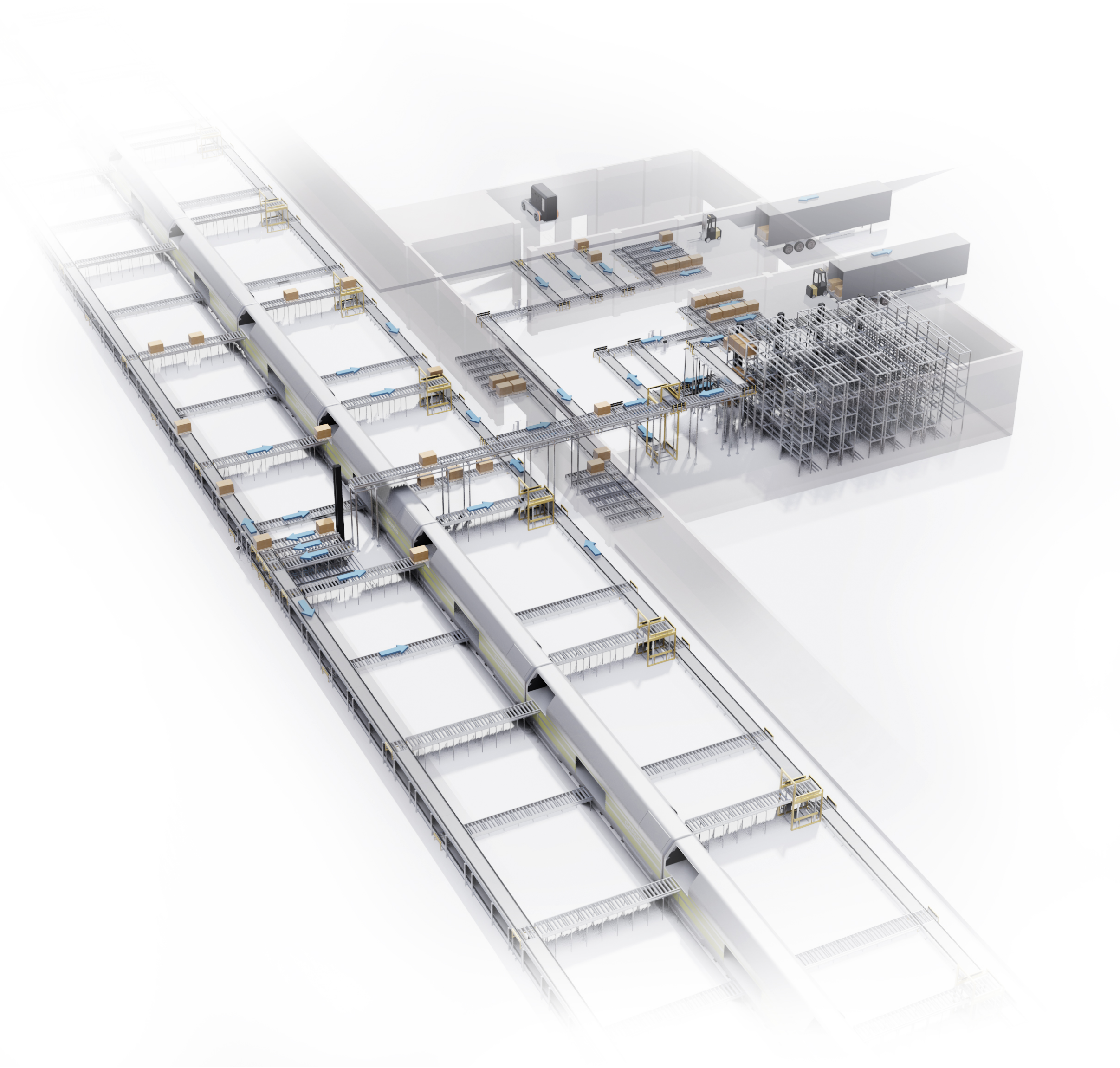


Figure 6-3: Digital rendering of the terminal core variant V01 (DLR)

Several aspects were brought up by the three-dimensional implementation. One major aspect was that the original terminal layout displayed in Figure 6-1 lacks in performance especially for long trains. This is due the location of the TAO (rail) at the head of the NGT CARGO train. LU for railcars at the end of the train have to be conveyed a long way. Yet, a displacement of the TAO (rail) brings along higher complexity in system design. Finally, it was decided to elevate the outbound branch (rail) of the ITS. Being set to a higher level in the building, the ITS outbound branch (rail) overpasses the railcars. This enabled a placing the TAO (rail) in the longitudinal center of the NGT CARGO train. Consequently, the LS for the NGT CARGO was extended to both sides of the TAO (Rail). To counteract potential performance issues, the main inbound roller conveyor of the ITS was doubled (as displayed in Figure 6-4). To feed the ITS outbound branch (rail), the WAI for outbound rail freight was elevated accordingly. The crossdocking transshipment with outbound rail direction was facilitated by adding another elevator.

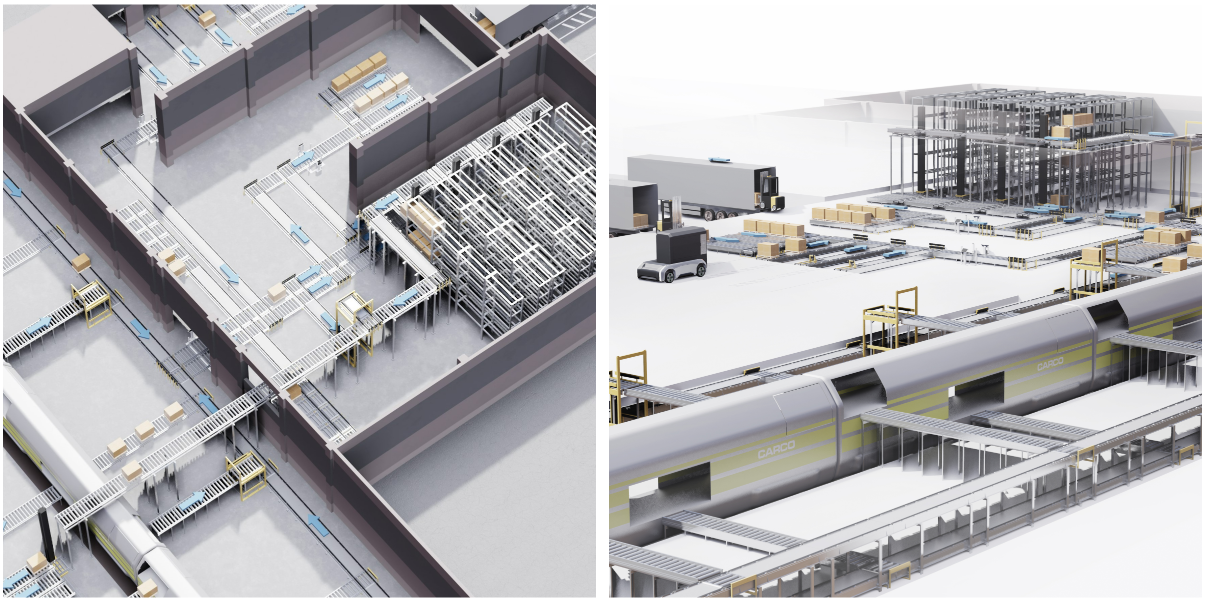


Figure 6-4: Detailed views of the rendered terminal core variant V01 (DLR)

A further aspect revealed within the process of digital rendering is an inefficient use of terminal space. However, a general plausibility of the terminal concept was confirmed, although further investigation on intralogistics performance is required.

Summarizing this section, the validation of the developed terminal architecture was basically successful. The overall concept was evaluated as being feasible and reasonable from a logistics perspective. Most of the requirements were satisfied by the product architecture or at least fulfilled on a process level. Yet, several requirements were not met, which implies that further investigation on their implementation is required. Possible approaches are material flow simulations to evaluate the performance requirements. Alternatively, rework on the product architecture with less simplifications regarding information flow or treatment of external interfaces seems to be promising to improve the validity. A visual rendering of the terminal was developed, putting the spot on plausibility and understanding. The visualization was experienced being suitable for communication issues or as discussion baseline. General plausibility of the terminal architecture was confirmed, yet subsequent validation or another logistics expert interview in regard to intralogistics performance is recommended.

6.2 Evaluation of Research Objectives

Baseline for this thesis was the research question, *how to apply MBSE successfully in order to guide the system architecture development of the NGT CARGO logistics terminal*. As key drivers to approach this question, two research objectives were defined in Section 1.3. In the following, these objectives are discussed in order to demonstrate the contribution of this thesis to the fundamental research question. Therefore, Section 6.2.1 evaluates the verified architecture concept (objective 1), while Section 6.2.2 focuses on MBSE as planning approach (objective 2).

6.2.1 Discussion of Research Objective 1

Objective 1: Develop an architecture of the NGT CARGO logistics terminal with focus on intralogistics freight handling.

The developed terminal system architecture is a framework consisting out of a base, functional, logical, and product architecture. According to the specification of the objective 1 in Section 1.3, the successful verification of the physical implementation of the terminal variant V01 is sufficient to meet the research objective 1. Yet, the realized product architecture represents only one variant of the NGT CARGO logistics terminal and bases on various assumptions and design decisions.

The basic assumptions met in Subsection 4.1 were found decisive for the successful architecture development. Covering 28 basic UC, a modeling of the whole terminal including site, vehicle infrastructure, or management operations would not have been feasible in the limited time. Further, the limiting to road vehicles as involved means of transport apart of the NGT CARGO was helpful. However, as long as the vehicle fulfills the system boundary interface condition (freight on a pallet; handover at the vehicle), any mean of transport can be addressed by selecting an appropriate implementation of the loading system without significantly changing the terminal's architecture.

Although the specification process of the terminal's architecture was rather comprehensive, the level of detail of the implemented product architecture is still rough. Yet, this was done on purpose with respect to the limited time frame of this thesis. Especially when specifying the intralogistics process (definition of UC activities, see Subsection 4.3.4), it was found hard not to slip into details. A lot of time was spent on the specification of this process, until an appropriate level of detail was found. Due to the focus on the material flow, the rather low level of detail for the essential activities did not interfere the derivation of the terminal system architecture.

In contrast, for a potential investigation on information flows, a more detailed specification of the essential activities would have been necessary. This becomes visible when looking at the secondary UC (see Figure 4-6), which are basically UC for information operations. Both are located on a lower level of process design and a detailed specification requires extensive work on the system's behavior in regard to information flows. In addition, the simplification of all information content as LU Info probably is not useful when investigating on information flows. However, in this thesis it has certainly contributed greatly to the successful creation of the architecture.

The decision to leave out the order picking process can be seen critically, as the order picking is a key process for added value in intralogistics [Hom-2018, p. 270]. If the order picking is included within the development scope, investigations on processes, implementation systems, and integration into the other material flow are required. This significantly drives up the complexity level of the development process as well as the terminal architecture itself. Hindsight, it was found as a useful simplification to realize a basic terminal system architecture in the limited time. The same applies to the exclusion of customs, although the resulting changes to the development process and architecture are likely to be much smaller.

A treatment of refused goods could be taken into account by implementing a graded decision guideline. For example, this guideline could imply rules whether a rejected LU still can be processed (minor errors, such as deviation in quantity), needs further investigation (medium errors, such as missing identity or quality issues) or has to be discharged to prevent damage from the intralogistics systems (major errors, such as damaged pallet or projecting LU shape). In that case, technical implementations for treating of LU with errors including holding facilities are required. Further, questions of legal status and ownership of freight arise and have to be considered.

However, the group of goods to handle was found to be the most important influencing factor on the terminal architecture. Apart from the technical complexity itself, a comprehensive knowledge about the individual requirements of the goods is required. The decision, to develop an architecture for cluster 2 only (see Section 4.6) was a major, but worthwhile simplification, as they do not require special treatment.

In comparison, transshipment of cluster 1 goods (e.g. fruits, vegetables) stipulates extensive requirements on temperature, humidity, and atmosphere (gases). As examples, every type of fruit exhibits an individual transport temperature with only marginal admissible variation, or not every type of fruit may be stored close by others for allelopathy reasons (e.g. citrus fruits and apples) [Ges-2021]. All this implies a string of changes on terminal architecture, such as implementation of condition control systems or spatially separated storage and buffering places for different goods. The implied cold warehouse differs in technical realization from variant V01, as interfaces with temperature gradients are difficult to realize with continuous conveyors or particular technology for cold storage has to be applied [Hom-2018, p. 102; Sch-2019b, p. 108].

Other groups of goods, such as cluster 4 (e.g. consolidated cargo, industry equipment), are transshipped using non-standardized pallets. An automated transshipment respectively storage was expected to be more complicated than in case of standardized Euro pallets. Depending on the dimensions of the goods, a use of an additional standardized in-house load carrier attached under the actual LU can simplify the handling of these goods and thus, the terminal architecture.

A final aspect with impact on the terminal architecture is its flexibility for expansion. In regard to the NGT CARGO operational concept, this can refer to parallel handling of more than one train with impact on transshipment and storage capacity. Alternatively, an adaption of track length for varying numbers of NGT railcars may be required depending on the operational role of the terminal (hub for block trains, logistics siding for few railcars).

The developed system architecture basically supports these adaptions by offering the logistics toolbox and providing all necessary components. Yet, to approach this flexibility strategically, a further definition of modules as standardized sets of logistics components is beneficial[3]. This can result in the pre-definition of terminal size variants (e.g. S, M, L,

[3] Here, the logical structure can serve as reference for standardized modules.

XL), differing in throughput or rail tracks. For example in case of minor track extension (e.g. S to M), a multiple implementation of the LS (rail) module is sufficient. This was partially considered in the graphical visualization of the product architecture (see Figure 6-1 and Figure 6-3). To display a NGT CARGO block train, the LS (rail) was extended using similar modules although the model itself considered one railcar only (see Figure 4-25).

However, the extension potential may be limited due to the performance of components interconnecting the modules creating congestion or tailback effects. Additionally, if the extension implies transshipment of additional groups of goods, potentially new requirements have to be considered as mentioned before.

Summarized, the design process revealed that the context of intermodal terminals is very complex and simplifications were needed to specify a consistent architecture within this thesis. However, the resulting system architecture represents a plausible concept for a particular application guided by an MBSE approach. Further, the developed system model enables a future creation of more variants for differing application cases. By fulfilling research objective 1, this thesis demonstrates that MBSE can be applied to guide the architecture development in context of an intermodal freight terminal.

6.2.2 Discussion of Research Objective 2

Objective 2: Evaluate the suitability of MBSE as a holistic approach to support the planning of logistics systems.

The previously presented successful validation gave evidence that the developed concept is plausible and feasible from a logistics perspective (see Section 6.1). Hence, it can be concluded that MBSE is suitable to successfully guide the planning of an intralogistics system. To meet research objective 2 completely, a concluding discussion of the MBSE approach in regard to its support of logistics planning is necessary.

In theory of developing logistics facilities, processes shall determine the structures and not vice versa [Gud-2010, p. 9]. This was also experienced within the modeling approach as the MBSE methodology proposes a fundamental system analysis in prior to the derivation of architectures. Especially when deriving the logical architecture, a clear definition of the process was found to be a great support. The means and tools provided by MBSE to support an initial definition of actions, sequences, and flows leading to the system processes were regarded as sufficient and helpful. The inherent top-down approach of MBSE supported a step-by-step familiarization with the logistics activities and stimulated a holistic systems thinking. This enabled a comprehensive understanding of the terminal's functionality, which was fundamental to the development of the structure. Especially on the behavioral level, the holistic and iterative characteristic of MBSE can be a great chance to cope problems in logistics planning, such as divisional thinking and the premature focusing on technical implementation [Hom-2018].

Apart from the top-down approach, MBSE encourages the creation of variants. The possibility to implement these variants on a behavioral or structural level within the model complies to the needs of logistics planning [Dur-2014]. Once the toolbox of logistics system variants is set up in the modeling tool, an easily applicable drag-and-drop of these system components facilitates a quick derivation of architecture variants. However, this was found suitable for conceptual purposes only. No automated checks for logistics compatibility among the applied system components were implemented. Further, reaching a certain quantity of components, the compilation of a product architecture was experienced as rather confusing and hard to visualize in a comprehensible way. MBSE respectively SysML and the Cameo Systems Modeler reach their limits, as this is more part of dedicated configuration tools. Although literature claims interconnectivity to be a major aspect for logistics planning [Dom-2018], the transformation from SysML model information to a configuration tool, quick plant modeling tool, or detailed CAD model for components was not covered by this thesis. Approaches exist, but these were not pursued due to constraints in time and scope [Kir-2017; Moe-2015].

During the modeling process in this thesis, MBSE was experienced especially valuable for its ease of use. By encouraging iterative modeling steps, the MBSE approach ensures that nothing is forgotten and thus, enhances the level of completeness. The comprehensive advice for the processes (what to do) and the methods (how to do) facilitated a target-oriented, effective progress. This is a major advantage, which might address the missing connection between the static logistics planning processes and the progress itself, as criticized by

Durchholz [Dur-2014].

In regard to consistency, the system model was found beneficial. Incompatible flows or processes as well as inappropriate block interfaces or other logical errors are prevented or at least reported by the modeling tool. As an example, the need for the sink for rejected LU was discovered by being notified about the inconsistent material flow between input and output of the UC activities (see Subsection 4.3.3). Despite this, without continuously maintaining the model, inconsistencies occur. The maintenance of the model was found to be cumbersome, in particular for late changes, requiring various adjustments in the whole model (e.g. the late implementation of the controlled condition flow in the logical architecture, see Figure 4-21).

Another found drawback of SysML with relevance for material flow planning is its strong focus on software and information. Already touched on in Section 4.5, the limit of SysML sequence diagrams to messages only proved difficulties in developing material flows. This was experienced as a needless limiting of the MBSE capabilities, as sequences are very important to identify the system's stakeholders respectively interacting components. Further, the strongly formalized characteristics of SysML enforce a bargaining between semantic correctness and comfort in readability. Especially in regard to the product architecture, the experience made in this thesis is that formally correct and complete MBSE views tend to be not suitable for communication with stakeholders (e.g. for validation purposes, see Section 6.1). Here, a selection of a different, less software-centered modeling language, such as ARCADIA, might be an improvement to this issue. A different approach is the derivation of a domain specific language as a SysML profile to enhance usability and comprehensibility of model views for domain specific purposes with focus on stakeholder communication [Mug-2020b]. Yet, the necessary effort must not be underestimated and must be in reasonable proportion to the benefits of domain-customized views.

Based on the experiences made in this thesis, Figure 6-5 shows a basic classification of the MBSE modeling steps taken within this thesis to current general logistics planning.

Covering basic project management tasks, the MBSE approach covers activities equivalent to the initial investigation phase. Here, especially the research on the targeted intralogistics process and the definition of the base architecture as system scope and boundary are

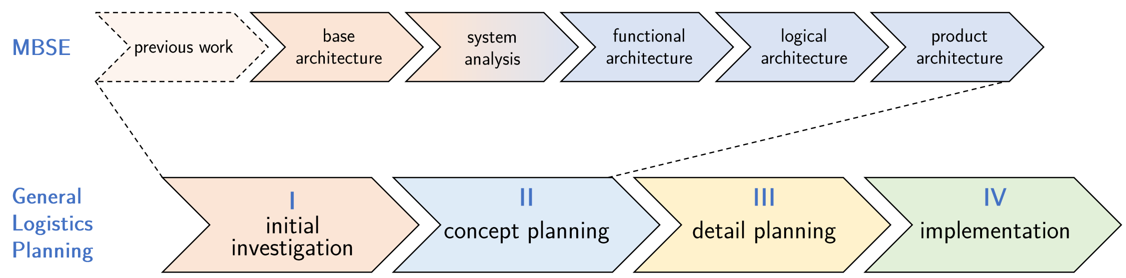


Figure 6-5: Potential classification of MBSE steps to current general logistics planning (own illustration based on Figure 2-22)

located. Further, the consistent evolution of the system model across several stages up to a rough layout as system variant is beneficial and equivalent to existing planning steps. Similarities, such as firstly determining functional units and their interrelations, as well as an initial neglecting of the spatial arrangement of components were detected in the concept stage of logistics planning. In the retrospective, MBSE can be evaluated to unfold great potential especially regarding the concept phase and is considered to be particularly valuable for increasing creativity and variability of the solution finding process. However, due to lack of experience in current logistics planning, no further detailed assessment of MBSE was possible within this thesis.

Concluding, MBSE can be seen as great chance to support the planning of logistics systems. In most of the aspects, the findings in regard to MBSE as guiding approach coincide with the conclusions from the literature research. Various benefits were experienced, addressing especially consistency and support in development progress. In contrast, problems in terms of model maintenance effort and readability of SysML views were noticed. Potential for integration of MBSE in current logistics planning is recognized, although MBSE is estimated not to be capable of replacing current logistics approaches completely. However, a combination of both, MBSE and the well-defined existing logistics planning processes seems to be promising. By fulfilling research objective 2, this thesis demonstrates, that MBSE is suitable to successfully support the planning of logistics systems in context of an intermodal freight terminal.

6.3 Discussion of Research Approach

In this final section of evaluation, the research approach conducted and chosen means are critically reflected.

The followed DRM framework was experienced especially beneficial to structure the thesis and to guide the evaluation process. In addition, it was useful to position the extensive modeling activities within a stringent and structured research approach. Beyond that, the DRM was perceived as rather academic and needless complicated.

This impression was mainly driven by the strength of MBSE, as it provides a comprehensive guidance in structuring the development process and tends to make other guiding superfluous. This guidance was a crucial part for the successful creation of the terminal system architecture within this thesis, as initially there was rather little background knowledge on the logistics domain. Here, especially the intensive literature research on the basic intralogistics process and the systems behavior were found valuable.

The subsequent detour in methodology via the functional architecture by integrating the FAS method was worthwhile, as this took the understanding of the terminal system to a next level. However, when designing the product architecture, lots of clear decisions in design were necessary. Due to the lack of logistics domain knowledge, this was experienced as rather hard. Hindsight, a more intensive feedback from logistics experts in regard to the design decisions is recommended.

Another issue occurring was a lack in project data for the NGT CARGO operational concept. The derivation of a terminal variant strongly depends on performance data such as throughput, material flow split ratio, quantity of railcars served simultaneously, frequency of trains, or storage duration. Missing information on these aspects made it hard to select the appropriate logistics systems from the toolbox. The vague description of the initial planning goal can be identified as one reason for performance issues of the product architecture, revealed by the validation. This issue confirmed literature findings, that an initial investigation phase with a comprehensive analysis of target data is crucial and would have needed special focus in the project approach.

The limited time frame of this project was found very challenging. This was mainly driven by the overall project complexity. Major simplifications had to be made for scope reasons. Thus, the resulting outcome can be criticized to be rather academic for omitting various aspects. However, the developed terminal system architecture was verified and validated. Hence, it basically fulfilled the expectations of the research approach targeted by this thesis.

The conducted interviews with the experts for validation and verification purposes were experienced as very supportive and helpful. As a part of the lessons learned, it can be stated that more expert interviews should be carried out. A more frequent and constant exchange with experts along the whole project (even in early phases) may had further increased the terminal system architecture's quality or validity within the same time frame.

Summarizing, the conducted research approach was successful as the research objectives were met and thus, an answer to the research question was given by this thesis. The selection of SYSMOD/FAS, SysML and the Cameo Systems Modeler as MBSE pillars was a good decision to facilitate the specification of the terminal system architecture. Apart from fundamental literature review, expert interviews supported the verification and validation of the resulting architecture. Embedded in the conducted research approach, MBSE has proven that it is suitable to find a good solution for unfamiliar terrain with a holistic and multidisciplinary approach.

7 Conclusion and Outlook

Within this final chapter, the thesis and its contribution for the initial problem is summarized. It further concludes this work by stating open aspects and giving possible questions for future research.

Conclusion

This thesis is embedded in the NGT CARGO logistics project by DLR. The NGT CARGO is a high-speed freight train concept designed for the transshipment of LDHV goods in an intermodal transportation chain. Within the project, the NGT CARGO logistics terminal is an essential part as it facilitates the intermodal transshipment of the goods among the NGT CARGO and road vehicles. A previously conducted system analysis of the logistics terminal revealed, that the existing preliminary conceptual terminal design is only partly capable of facilitating the required logistics processes and strongly depends on the type of goods.

The intention of this thesis was to apply an MBSE approach to develop a detailed and realistic system architecture of the NGT CARGO logistics terminal. Therefore, two research objectives were defined. These objectives targeted the specification of the architecture with focus on intralogistics freight handling as well as an evaluation of the suitability of MBSE to support such a logistics planning process.

Literature research was conducted to understand the fundamental principles of MBSE, intermodal terminals, and intralogistics freight handling. Further, examination of state-of-the-art logistics planning showed that various processes exist, yet they exhibit certain challenges such as divisional thinking or difficulties in application of digital support. Here, MBSE was found promising to contribute to a more consistent and comprehensive approach. A review of up-to-date modeling languages, methodologies, and tools including selection advice was carried out to facilitate the design of an appropriate MBSE approach.

The selected MBSE approach was a combination of SysML, Cameo Systems Modeler, and a tailored SYSMOD approach, including the FAS method. Guided by this tailored procedure, a system analysis of the terminal core was conducted to define the intralogistics processes as basic behavior of the terminal. Based on these findings, the terminal's generic structure was modeled, including a base, functional, and logical architecture. A toolbox with logistics system components including selection criteria was implemented to enable the creation of terminal variants. The final outcome was a verified product architecture variant as one possible physical implementation of the terminal specifically designed for selected groups of goods.

Evaluation certified that the developed terminal variant was both, plausible and reasonable from a logistics perspective. Regarding the architecture, the high level of complexity and the strong dependency on the individual good were managed by modular variant design and focus on homogeneous goods. The discussion of the MBSE approach confirmed the conclusions from literature review and attested MBSE a high potential to support the concept phase of current logistics planning.

In a nutshell, this thesis demonstrates how MBSE can be successfully applied to guide the system architecture development of an intermodal freight terminal with focus on the intralogistics freight handling.

Outlook

The complexity of future intermodal freight terminals is challenging to logistics planning. This thesis delivered a possible approach to manage this complexity by the application of MBSE. As a main outcome, a product architecture variant of the NGT CARGO logistics terminal was specified within a system model. Yet, several topics were identified as desirable thematic continuation of this work.

Initially, the developed system model should be further specified. The modeled structure can be allocated to its behavior on a component level, so that each logistics component actually exhibits different states and executes actions. Further, corresponding parametrics should be implemented and assigned to the components to address performance issues and facilitate more valid consistency evaluation for material flows. Examples for such parametrics could be throughput [LU/h] per conveyor or capacity values for storage facilities. Alternatively, the existing architecture can be further enriched by rework focusing on information flows. The implemented logistics toolbox should be used to derive further terminal variants. These variants may target various groups of goods, other means of transportation, or different terminal sizes.

Regarding a further use of MBSE in logistics planning, it is advisable to create a stereotype profile such as the SYSMOD profile used in this thesis. With such a profile, SysML could be applied more efficiently for logistics purpose. The derivation of a logistics meta model would be even more beneficial, yet it requires a lot of expertise and effort. Once it is implemented, this meta model should contribute significantly to a quick acceptance of MBSE in logistics planning among participants and stakeholders. However, as MBSE revealed high potential for enhancing creativity and consistency in complex architecture design, a further application in logistics planning projects should be promoted.

The NGT CARGO project should take advantage of the architecture specification of the terminal resulting by this work. Within the next project phase, a material flow simulation of the developed terminal variant should be carried out using modeling software such as Modelica, Anylogic, or Matlab Simulink. By doing so, a successful, comprehensive validation of the performance requirements can be achieved. This will further sharpen the concept of the NGT CARGO logistics terminal as part of a future-oriented flagship project for sustainable intermodal freight transport.

Although the NGT CARGO is still a concept, the relevance of high-speed rail freight is already noticeable today. In December 2020, the China Railway Rolling Stock Corporation (CRRC) Tangshan presented the world's first 350 km/h freight train, which is comparable to the NGT CARGO concept [Xin-2020]. In addition, the demand for trans-continental long haul rail freight along the 'silk road' is growing [Arn-2019b]. On the other hand, issues like the blocking of the Suez Canal by the container ship Ever Given in March 2021 reveal the vulnerability of current logistics chains [Yee-2021]. Together with new opportunities triggered by the current pandemic crisis [ORF-2021], high-speed rail freight seems to be promising and is an exciting field of research where MBSE might evolve as major factor for success.

Bibliography

[Aßm-2019] Aßmann, R.: Systemtechnik für die Stückgutförderung. In: Innerbetriebliche Logistik. Schmidt, T. (editor). Fachwissen Logistik. Springer Vieweg, Berlin, 2019, pp. 1–40.

[Ali-2006] Alicke, K.; Lippolt, C.; Wisser, J.: Prozessorientiertes Benchmarking von Distributionszentren in Wertschöpfungsnetzwerken. In: Intralogistik. Potentiale, Perspektiven, Prognosen. Arnold, D. (editor). VDI-Buch. Springer-Verlag GmbH, Berlin Heidelberg, 2006, pp. 91–100.

[Alt-2012] Alt, O.: Modellbasierte Systementwicklung mit SysML. Hanser, München, 2012.

[Arn-2006] Arnold, D.: Einleitung des Herausgebers. In: Intralogistik. Potentiale, Perspektiven, Prognosen. Arnold, D. (editor). VDI-Buch. Springer-Verlag GmbH, Berlin Heidelberg, 2006, pp. 1–4.

[Arn-2019a] Arnold, D.; Furmans, K.: Materialfluss in Logistiksystemen. 2019.

[Arn-2019b] Arnz, M.; Böhm, M.; Weibezahn, j.: Mit Hochgeschwindigkeit auf der Seidenstraße. Ergebnisse einer Potenzialanalyse des Transportkorridors Shanghai – Duisburg. In: Internationales Verkehrswesen, vol. 71 (2019) no. 3, pp. 38–41.

[ATS-2021] ATS Group: Pallet conveyor system workshop. 2021. Url: https://www.ats-group.com/EN/product-solutions/pallet-conveyor-system.html (visited on 03/05/2021).

[Bad-2018] Badache, N.; Roques, P.: Capella to SysML Bridge: A Tooled-up Methodology for MBSE Interoperability. Toulouse: 9th European Congress on Embedded Real Time Software and Systems, 2018.

[Bah-1998] Bahill, A.; Gissing, B.: Re-Evaluating Systems Engineering Concepts Using Systems Thinking. In: IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 28 (1998) no. 4, pp. 516–527.

[Bek-2007] Bektas, T.; Crainic, T. G.: A Brief Overview of Intermodal Transportation. Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT), Montreal, 2007.

[Ber-1968] Bertalanffy, L. von: General System Theory: Foundations, Development, Applications. Braziller, New York, 1968.

[Beu-2019] Beumer, C.; Jodin, D.: Sortier- und Verteilsysteme. In: Innerbetriebliche Logistik. Schmidt, T. (editor). Fachwissen Logistik. Springer Vieweg, Berlin, 2019, pp. 153–174.

[Ble-2009] Blessing, L. T.; Chakrabarti, A.: DRM, a Design Research Methodology. Springer London, London, 2009.

[Böh-2017] Böhm, M.; Malzacher, G.: Funktionales Lastenheft NGT CARGO. Deutsches Zentrum für Luft- und Raumfahrt e.V., Institut für Fahrzeugkonzepte, 2017.

[Böh-2019] Böhm, M.; Malzacher, G.; Münster, M.; Winter, J.: NGT Logistics Terminal. Ein Güterumschlagkonzept für die intermodale Vernetzung von Schiene und Straße. In: Internationales Verkehrswesen, vol. 71 (2019) no. 1, pp. 38–41.

[Bon-2000] Bontekoning, Y. M.: The Importance of New-Generation Freight Terminals For Intermodal Transport. In: Journal of Advanced Transportation, vol. 34 (2000) no. 3, pp. 391–413.

[Bra-2020] Brandtstätter, M.: Advanced Systems Engineering. Lecture notes. Lehrstuhl für Raumfahrttechnik, Technische Universität München, Garching, 2020.

[Bra-2021] Brandtstätter, M.: Lehrstuhl für Raumfahrttechnik, Technische Universität München, Garching. Expert Interview on Feb. 10, 2021.

[Bun-2016] Bundesministerium für Verkehr und digitale Infrastruktur: Bundesverkehrswegeplan 2030. 2016.

[Bun-2017] Bundesministerium für Verkehr und digitale Infrastruktur: Masterplan Schienengüterverkehr. Berlin, 2017.

[Car-2008] Caris, A.; Macharis, C.; Janssens, G. K.: Planning Problems in Intermodal Freight Transport: Accomplishments and Prospects. In: Transportation Planning and Technology, vol. 31 (2008) no. 3, pp. 277–302.

[Cas-2017] Casse, O.: SysML in action with Cameo systems modeler. Implementation of model based system engineering set. ISTE Press Ltd, London, 2017.

[Clo-2010] Cloutier, R. J.; Bone, M.: Compilation of SysML RFI - Final Report. Systems Modeling Language (SysML) Request for Information OMG Document: syseng/2009-06-01. Hoboken, NJ: Stevens Institute of Technology, 2010.

[Dae-1977] Daenzer, W. F., ed.: Systems Engineering. Leitfaden zur Methodischen Durchführung Umfangreicher Planungsvorhaben. Verlag Industrielle Organisation, Zürich, 1977.

[Dae-2014] Daenzer, M.; Kleiner, S.; Lamm, J. G.; Moeser, G.; Morant, F.; Munker, F.; Weilkiens, T.: Funktionale Systemmodellierung nach der FAS-Methode: Auswertung von vier Industrieprojekten. In: Tag des Systems Engineering 2014. Maurer, M.; Schulze, S.-O. (editor). Carl Hanser Verlag GmbH & Co. KG, München, 2014, pp. 75–84.

[Del-2014] Delligatti, L.: SysML distilled. A brief guide to the systems modeling language. Addison-Wesley, Upper Saddle River, NJ, 2014.

[Deu-2017] Deutsche Bahn Netz AG: MegaHub Lehrte - Projektvorstellung im Rahmen des Info-Tages. Unverbindlicher Planungsstand April 2017. Hannover, Apr. 26, 2017.

[Dic-2017] Dickopf, T.; Schulte, T.; Schneider, M.: Analyse existierender SysML-basierter Ansätze aus Industrie und Forschung. In: Modellbasierter Entwicklungsprozess cybertronischer Systeme. Der PLM-unterstützte Referenzentwicklungsprozess für Produkte und Produktionssysteme. Eigner, M.; Koch, W.; Muggeo, C. (editor). Springer Vieweg, Berlin, 2017, pp. 65–72.

[DIN-30781] Deutsches Institut für Normung e.V., ed.: Transportkette - Grundbegriffe. DIN 30781. Berlin, 1989.

[Dom-2018] Dombrowski, U.; Ernst, S.; Reimer, A.: Fabrikplanung. In: KlimaIng - Planung klimagerechter Fabriken. Problembasiertes Lernen in den Ingenieurwissenschaften. Dombrowski, U.; Marx, S. (editor). Springer Vieweg, Berlin, 2018, pp. 51–80.

[Dor-2002] Dori, D.; Crawley, E. F.: Object-Process Methodology. A Holistic Systems Paradigm. Springer, Berlin, 2002.

[Dur-2014] Durchholz, J.: Vorgehen zur Planung eines schlanken Logistikprozesses. Wertstromdesign für die Logistik. Lehrstuhl für Fördertechnik Materialfluss Logistik. Dissertation. München: Technischen Universität München, 2014.

[Ehr-2020] Ehret, M.; Böhm, M.; Malzacher, G.; Popa, A.: System Analysis of a High-Speed Freight Train Terminal. In: 23rd IEEE International Conference on Intelligent Transportation Systems (Rhodes, Greece). IEEE (editor). 2020, pp. 3360–3365.

[Eig-2017] Eigner, M.: Ausgangssituation. In: Modellbasierter Entwicklungsprozess cybertronischer Systeme. Der PLM-unterstützte Referenzentwicklungsprozess für Produkte und Produktionssysteme. Eigner, M.; Koch, W.; Muggeo, C. (editor). Springer Vieweg, Berlin, 2017, pp. 5–12.

[Est-2008] Estefan, J. A.: Survey of Model-Based Systems Engineering (MBSE) Methodologies. INCOSE MBSE Initiative, Pasadena, CA, 2008.

[Eur-2011] European Commission: WHITE PAPER. Roadmap to a single European transport area - Towards a competitive and resource efficient transport system. Brussels, 2011. Url: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:52011DC0144 (visited on 02/14/2021).

[Eur-2012] European Commission: European Rail Traffic Management System and the 2007-2013 TEN-T Programme + Corridor implementation. Brussels, 2012.

[Eur-2015] European Commission: Commission Implementing Regulation (EU) 2015/1998. 2015.

[Eur-2019b] European Commission: The European Green Deal sets out how to make Europe the first climate-neutral continent by 2050, boosting the economy, improving people's health and quality of life, caring for nature, and leaving no one behind. Brussels, Dec. 11, 2019. Url: https://ec.europa.eu/commission/presscorner/detail/en/ip_19_6691 (visited on 02/14/2021).

[Eur-2020] European Commission: The journey begins – 2021 is the European Year of Rail! Brussels, Dec. 30, 2020. Url: https://ec.europa.eu/commission/presscorner/detail/en/IP_20_2528 (visited on 02/14/2021).

[Eur-2021] European Pallet Association e.V.: Load Carriers Overview. 2021. Url: https://www.epal-pallets.org/eu-en/load-carriers/overview (visited on 03/02/2021).

[Fei-2020] Feiner, L.: Lehrstuhl für Fördertechnik Materialfluss Logistik, Technische Universität München, Garching. Expert Interview on Nov. 20, 2020.

[Fei-2021] Feiner, L.: Lehrstuhl für Fördertechnik Materialfluss Logistik, Technische Universität München, Garching. Expert Interview on Feb. 19, 2021.

[Fel-2021] Fellenberg, R.: Europa macht auf der Schiene mobil. In: VDI nachrichten, vol. 75 (2021) no. 4, p. 10.

[Flo-1993] Flood, R. L.; Carson, E. R.: Dealing with complexity. An introduction to the theory and application of systems science. Plenum Press, New York, 1993.

[For-1991] Forsberg, K.; Mooz, H.: The Relationship of System Engineering to the Project Cycle. In: INCOSE International Symposium, vol. 1 (1991) no. 1, pp. 57–65.

[Fot-2020] Fottner, J.: Planung technischer Logistiksysteme. Lecture notes. Lehrstuhl für Fördertechnik Materialfluss Logistik, Technische Universität München, Garching, 2020.

[Fri-2014] Friedenthal, S.; Moore, A.; Steiner, R.: A Practical Guide to SysML. The Systems Modeling Language. Morgan Kaufmann, Boston, MA, 2014.

[GEB-2021] GEBHARDT Fördertechnik GmbH: GEBHARDT pallet conveyor technology. 2021. Url: https://www.gebhardt-foerdertechnik.de/en/products/pallet-transport-technology/ (visited on 03/15/2021).

[Ges-2021] Gesamtverband der Deutschen Versicherungswirtschaft e.V.: Container Handbook. Cargo loss prevention information from German marine insurers. 2021. Url: http://www.containerhandbuch.de/chb_e/scha/index.html (visited on 03/25/2021).

[Gud-2010] Gudehus, T.: Logistik. Grundlagen - Strategien - Anwendungen. Springer, Berlin, 2010.

[Gün-2010] Günthner, W.; Hompel, M. ten, eds.: Internet der Dinge in der Intralogistik. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[Hab-2019] Haberfellner, R.; Weck, O. L. de; Fricke, E.; Vössner, S.: Systems Engineering. Fundamentals and Applications. Birkhäuser, Cham, 2019.

[Hal-1962] Hall, A. D.: A Methodology of Systems Engineering. Van Nostrand, Princeton, NJ, 1962.

[Hof-2011] Hoffmann, H.-P.: Deskbook Release 3.1.2. Model-Based Systems Engineering with Rational Rhapsody and Rational Harmony for Systems Engineering. IBM Corporation, Somers, NY, 2011.

[Hol-2013] Holt, J.; Perry, S.: SysML for systems engineering. A model based approach. Institution of Engineering and Technology, London, 2013.

[Hom-2011] Hompel, M. ten; Heidenblut, V.: Taschenlexikon Logistik. Abkürzungen, Definitionen und Erläuterungen der wichtigsten Begriffe aus Materialfluss und Logistik. VDI-Buch. Springer, Berlin, 2011.

[Hom-2018] Hompel, M. ten; Schmidt, T.; Dregger, J.: Materialflusssysteme. Förder- und Lagertechnik. VDI-Buch. Springer Vieweg, Berlin, 2018.

[Hom-2019] Hompel, M. ten; Sadowsky, V.; Mühlenbrock, S.: Kommissioniersysteme. In: Innerbetriebliche Logistik. Schmidt, T. (editor). Fachwissen Logistik. Springer Vieweg, Berlin, 2019, pp. 113–152.

[Hug-1998] Hughes, T. P.: Rescuing Prometheus. Pantheon Books, New York, 1998.

[IEE-1220] IEEE 1220-2005, ed.: Standard for Application and Management of the Systems Engineering Process. Institute of Electrical and Electronics Engineers (IEEE), 2005.

[INC-2014] INCOSE: A World in Motion – Systems Engineering Vision 2025. San Diego, CA: International Council on Systems Engineering, 2014. Url: https://www.incose.org/products-and-publications/se-vision-2025 (visited on 12/03/2020).

[INC-2015] INCOSE: Systems Engineering Handbook. A Guide for System Life Cycle Processes and Activities. Wiley, Hoboken, NJ, 2015.

[Isl-2018] Islam, D. M. Z.; Zunder, T. H.: Experiences of rail intermodal freight transport for low-density high value (LDHV) goods in Europe. In: European Transport Research Review, vol. 10 (2018) no. 24.

[ISO-15288] International Organization for Standardization, ed.: Systems and Software Engineering - System Life Cycle Processes. ISO/IEC/IEEE 15288:2015. Geneva, 2015.

[ISO-19450] International Organization for Standardization, ed.: Automation Systems and Integration - Object-Process Methodology. ISO/PAS 19450:2015. Geneva: International Organization for Standardization, 2015.

[ISO-19514] International Organization for Standardization, ed.: Information Technology - Object Management Group Systems Modeling Language (OMG SysML). ISO/IEC 19514:2017. Geneva: International Organization for Standardization, 2017.

[ISO-42010] International Organization for Standardization, ed.: Systems and Software Engineering - Architecture Description. ISO/IEC/IEEE 42010:2011. Geneva, 2011.

[Jac-2014] Jackson R; Islam DMZ; Zunder TH; Schoemaker J; Dasburg N: A market analysis of the low density high value goods flow in Europe. In: 13th World Conference on Transport Research (WCTR), (2014).

[Jet-2007] Jetzke, S.: Grundlagen der modernen Logistik. Methoden und Lösungen. Hanser, München, 2007.

[Jon-2011] Jong, K. de; Spain, M. de; Hernandez, M.; Post, D.; Taylor, J.: Process for Selecting Engineering Tools – Applied to Selecting a SysML Tool. Albuquerque: Sandia National Laboratories, 2011.

[Kil-2008] Kille, C.; Schmidt, N.: Wirtschaftliche Rahmenbedingungen des Güterverkehrs. Studie zum Vergleich der Verkehrsträger im Rahmen des Logistikprozesses in Deutschland. Fraunhofer IRB Verl., Stuttgart, 2008.

[Kir-2017] Kirsch, L.; Muggeo, C.; Schulte, T.; Schneider, M.; Müller, P.: PLM-Funktionen im Kontext von Systemmodellen. In: Modellbasierter Entwicklungsprozess cybertronischer Systeme. Der PLM-unterstützte Referenzentwicklungsprozess für Produkte und Produktionssysteme. Eigner, M.; Koch, W.; Muggeo, C. (editor). Springer Vieweg, Berlin, 2017, pp. 169–176.

[Kle-2013] Kleiner, S.; Kramer, C.: Model Based Design with Systems Engineering Based on RFLP Using V6. In: Smart Product Engineering. Proceedings of the 23rd CIRP Design Conference, Bochum, Germany, March 11th - 13th, 2013. Abramovici, M.; Stark, R. (editor). Lecture Notes in Production Engineering. Springer, Berlin, 2013, pp. 93–102.

[Lam-2006] Lampe, H.: Untersuchung von Dispositionsentscheidungen in Umschlagterminals des kombinierten Verkehrs Schiene-Straße. Logistik, Verkehr und Umwelt. Verlag Praxiswissen, Dortmund, 2006.

[Lam-2014] Lamm, J. G.; Weilkiens, T.: Method for Deriving Functional Architectures from Use Cases. In: Systems Engineering, vol. 17 (2014) no. 2, pp. 225–236.

[Lan-2019] Lange, A.-K.; Kastner, M.: KV-Terminals dynamisch planen. In: LT-manager, vol. 9 (2019) no. 47, pp. 26–29.

[Lyk-2000] Lykins, H.; Friedenthal, S.; Meilich, A., eds.: Adapting UML for an Object-Oriented Systems Engineering Method (OOSEM). vol. 10. INCOSE International Symposium. Wiley, Minneapolis, 2000.

[Mad-2018] Madni, A. M.; Sievers, M.: Model-Based Systems Engineering: Motivation, Current Status, and Research Opportunities. In: Systems Engineering, vol. 21 (2018) no. 3, pp. 172–190.

[Mal-2020] Malzacher, G.; Ehret, M.; Böhm, M.; Popa, A.: Systemanalyse für ein Güterverkehrsterminal. Anwendung des Model-Based System Engineering im Kontext des Next Generation Train CARGO. In: Internationales Verkehrswesen, vol. 72 (2020) no. 3, pp. 72–77.

[Mar-2016] Martin, H.: Transport- und Lagerlogistik. Systematik, Planung, Einsatz und Wirtschaftlichkeit. Springer Vieweg, Wiesbaden, 2016.

[Mei-2013] Meier, F.; Sender, J.; Voll, R.: Schienengüterverkehr. In: Verkehrs- und Transportlogistik. Clausen, U.; Geiger, C. (editor). Springer Vieweg, Berlin, 2013, pp. 161–177.

[Moe-2015] Moeser, G.; Kramer, C.; Grundel, M.; Neubert, M.; Kümpel, S.; Scheithauer, A.; Kleiner, S.; Albers, A.: Fortschrittsbericht zur modellbasierten Unterstützung der Konstrukteurstätigkeit durch FAS4M. In: Tag des Systems Engineering 2015. Schulze, S.-O.; Muggeo, C. (editor). Carl Hanser Fachbuchverlag, München, 2015, pp. 69–78.

[Mon-2017a] Monios, J.; Bergqvist, R., eds.: Intermodal Freight Transport & Logistics. CRC Press, Boca Raton, FL, 2017.

[Mon-2017b] Monios, J.; Bergqvist, R.: Introduction. In: Intermodal Freight Transport & Logistics. Monios, J.; Bergqvist, R. (editor). CRC Press, Boca Raton, FL, 2017, pp. 3–15.

[Mön-2020] Mönsters, M.; Flamm, L.: Operational Concept for the NGT CARGO on the European Reference Route Madrid – Bukarest. In: SIGNAL+DRAHT, vol. 112 (2020) no. 4, pp. 27–34.

[Mug-2020a] Muggeo, C.; Weilkiens, T.: The MBSE Podcast. Episode 2 – Die Geschichte der SysML, Episode of Nov. 05, 2020. 2020. Url: https://mbse-podcast.rocks/2-die-geschichte-der-sysml-de/ (visited on 02/25/2021).

[Mug-2020b] Muggeo, C.; Weilkiens, T.: The MBSE Podcast. Episode 4 - Dependable System-of-Systems Engineering. Guest: Prof. Dr. Christian Neureiter; Episode of Dec. 15, 2020, 00:16:20-00:26:45. 2020. Url: https://mbse-podcast.rocks/episode-4-dependable-system-of-systems-engineering-with-christian-neureiter-en/ (visited on 12/16/2020).

[Mug-2021] Muggeo, C.; Weilkiens, T.: The MBSE Podcast. Episode 6 - MBSE im Maschinenbau. Guests: Dr. Jörg Berroth, Dr. Michael Riesener, Episode of Jan. 26, 2021, 00:23:07-00:24:20. 2021. Url: https://mbse-podcast.rocks/episode-6-mbse-im-maschinenbau-mit-dr-jorg-berroth-und-dr-michael-riesener-de/ (visited on 02/20/2021).

[Mül-1997] Müller, W.: Planung, Bau und Betrieb von Umschlagbahnhöfen in Deutschland. In: Eisenbahntechnische Rundschau, vol. 46 (1997) no. 10, pp. 624–628.

[Mur-2001] Murray, C.: RUP SE: The Rational Unified Process for Systems Engineering. Rational Software, 2001.

[Mus-2013] Muschkiet, M.; Ebel, G.: Begriffe und Systematik. In: Verkehrs- und Transportlogistik. Clausen, U.; Geiger, C. (editor). Springer Vieweg, Berlin, 2013, pp. 123–136.

[Nat-1993] National Institute of Standards and Technology, ed.: Standard for Integration Definition for Function Modeling (IDEF0). Draft Federal Information Processing Standards Publication 183, 1993.

[Net-2010] Nettsträter, A.; Kuzmany, F.: Rechenplattformen und RFID für das Internet der Dinge. In: Internet der Dinge in der Intralogistik. Günthner, W.; Hompel, M. ten (editor). Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 107–118.

[NoM-2021] NoMagic: Cameo Systems Modeler. 2021. Url: https://www.nomagic.com/products/cameo-systems-modeler (visited on 03/10/2021).

[Obj-2017] Object Management Group: OMG Unified Modeling Language (OMG UML). Version 2.5. 2017. Url: https/www.omg.org/spec/UML/ (visited on 12/04/2020).

[Obj-2019] Object Management Group: OMG Systems Modeling Language (OMG SysML). Version 1.6. 2019. Url: https://www.omg.org/spec/SysML/1.6/ (visited on 12/01/2020).

[ORF-2021] ORF.at: Bahn statt Schiff. Pandemie gibt „Seidenstraße“ neuen Schub. 2021. Url: https://orf.at/stories/3197149/ (visited on 02/14/2021).

[Par-2010] Paredis, C. J.; Bernard, Y.; Burkhart, R. M.; Koning, H.-P. de; Friedenthal, S.; Fritzson, P.; Rouquette, N. F.; Schamai, W.: An Overview of the SysML-Modelica Transformation Specification. In: INCOSE International Symposium, vol. 20 (2010) no. 1, pp. 709–722.

[Piv-2021] PivotPoint Technology Corporation: Commercial, Free & Open Source SysML Modeling Tools. 2021. Url: https://sysml.org/sysml-tools/ (visited on 03/10/2021).

[Ram-2012] Ramos, A. L.; Ferreira, J. V.; Barceló, J.: Model-Based Systems Engineering: An Emerging Approach for Modern Systems. In: IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42 (2012) no. 1, pp. 101–111.

[Roq-2016] Roques, P.: MBSE with the ARCADIA Method and the Capella Tool. 2016. Url: https://api.semanticscholar.org/CorpusID:9283412 (visited on 12/05/2020).

[Ros-2018] Rosenow, H.: Trade Off Bewertungsmethodik für Tool- und Methodeentscheidungen zur Virtualisierung und Modellbasierung in der Entwicklung. Master's Thesis. Lehrstuhl für Raumfahrttechnik, Technische Universität München, Garching, 2018.

[Roy-1970] Royce, W. W.: Managing the Development of Large Software Systems. In: Proceedings IEEE WESCON, (1970), pp. 1–9.

[Sch-2019a] Schmidt, T., ed.: Innerbetriebliche Logistik. Fachwissen Logistik. Springer Vieweg, Berlin, 2019.

[Sch-2019b] Schmidt, T.; Hahn-Woernle, P.; Heptner, F.: Lagersysteme für Stückgut. In: Innerbetriebliche Logistik. Schmidt, T. (editor). Fachwissen Logistik. Springer Vieweg, Berlin, 2019, pp. 73–112.

[Sch-2020] Schroeder, P.: Zügig zum Ziel. In: VDI nachrichten, vol. 74 (2020) no. 49, p. 11.

[SEB-2020] SEBoK Editorial Board: The Guide to the Systems Engineering Body of Knowledge (SEBoK), v2.3, R.J. Cloutier (Editor in Chief). Cloutier, R. J. (editor). Hoboken, NJ: The Trustees of the Stevens Institute of Technology, 2020.

[Sta-1973] Stachowiak, H.: Allgemeine Modelltheorie. Springer, Wien, 1973.

[Ste-2015] Steiner, R.: The Four Pillars of SysML. Rebuilding the original 2006 graphics for clarity. Skygazer Consulting. 2015. Url: https://www.youtube.com/watch?v=998UznK9ogY (visited on 02/23/2021).

[Uni-2001] United Nations; Economic Commission for Europe: Terminology on Combined Transport. New York and Geneva, 2001.

[VDI-2489] Verein Deutscher Ingenieure, ed.: Procedure in material flow planning. VDI directive 2489. 2011.

[VDI-3590] Verein Deutscher Ingenieure, ed.: Order-picking systems. VDI directive 3590. 1994.

[VDI-3619] Verein Deutscher Ingenieure, ed.: Sorting and distribution systems for parcel goods. VDI directive 3619. 2017.

[VDI-3637] Verein Deutscher Ingenieure, ed.: Data collection for long term factory planning. VDI directive 3637. 1996.

[VDI-3694] Verein Deutscher Ingenieure, ed.: System requirement/specification for planning and design of automation systems. VDI directive 3694. 2014.

[VDI-5200] Verein Deutscher Ingenieure, ed.: Factory planning - Planning procedures. VDI directive 5200. 2011.

[Voi-2018] Voirin, J.-L.: Model-based System and Architecture Engineering with the Arcadia Method. eng. Implementation of model based system engineering set. Elsevier and ISTE Press, Kidlington, Oxford and London, 2018.

[Wal-2018] Walter, U.: Systems Engineering. Lecture notes. Lehrstuhl für Raumfahrttechnik, Technische Universität München, Garching, 2018.

[Wei-2014] Weilkiens, T.: Systems Engineering mit SysML/UML. Anforderungen, Analyse, Architektur. dpunkt.verl., Heidelberg, 2014.

[Wei-2016a] Weilkiens, T.; Scheithauer, A.; Di Maio, M.; Klusmann, N.: Evaluating and Comparing MBSE Methodologies for Practitioners. In: International Symposium on Systems Engineering. Proceedings Papers. IEEE (editor). IEEE, George Hotel, Piscataway, NJ, 2016, pp. 1–8.

[Wei-2016b] Weilkiens, T.: SYSMOD - The Systems Modeling Toolbox. Pragmatic MBSE with SysML. 2nd edition, version 4.1. MBSE4U booklet series. MBSE4U, Fredesdorf, 2016.

[Wei-2016c] Weilkiens, T.; Lamm, J. G.; Roth, S.; Walker, M.: Model-Based System Architecture. Wiley series in systems engineering and management. Wiley, Hoboken, NJ, 2016.

[Wei-2016d] Weimer, J.; Schmid, S.; Schier, M.; Rinderknecht, F.; Bünte, T.: Next Generation Car – Technologies for future EVs. Montréal, Québec, 2016.

[Wei-2020a] Weigand, W.: Klimaziele erreichen: Ein neues Produktionssystem im kombinierten Verkehr kann den Modal-Split signifikant verändern. In: Eisenbahntechnische Rundschau, vol. 69 (2020) no. 9, pp. 20–25.

[Wei-2020b] Weilkiens, T.: MBSE4U. Popular SysML/MBSE Modeling Tools. 2020. Url: https://mbse4u.com/sysml-tools/ (visited on 12/05/2020).

[Wei-2021] Weilkiens, T.: MBSE4U. SysML Reference Cards. 2021. Url: https://model-based-systems-engineering.com/wp-content/uploads/2012/03/sysmod-sysml-1.3-reference-card-weilkiens.pdf (visited on 03/26/2021).

[Wid-2017] Widong, M.; Leisch, N.: Neuer multimodaler Umschlagbahnhof für Luxemburg. In: Eisenbahntechnische Rundschau, vol. 66 (2017) no. 9, pp. 40–45.

[Win-2017a] Winter, J.; Böhm, M.; Malzacher, G.; Krüger, D.: NGT CARGO – Schienengüterverkehr der Zukunft. In: Internationales Verkehrswesen, vol. 69 (2017) no. 2, pp. 82–85.

[Win-2017b] Winter, J.; Krüger, D.; Böhm, M.; Mönsters, M.; Schumann, T.: Ein Betriebskonzept für den internationalen Güterverkehr. In: Deine Bahn, vol. 45 (2017) no. 8, pp. 12–16.

[Wis-2009] Wisser, J.: Der Prozess Lagern und Kommissionieren im Rahmen des Distribution Center Reference Model (DCRM). Zugl.: Karlsruhe, Univ., Diss., 2009. vol. 72. Wissenschaftliche Berichte des Institutes für Fördertechnik und Logistiksysteme der Universität Karlsruhe (TH). Universitätsverlag, Karlsruhe, 2009.

[Wor-2020] World Health Organisation: WHO Director-General's opening remarks at the Media briefing on COVID-19. Mar. 11, 2020. Url: https://www.who.int/director-general/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19—11-march-2020 (visited on 02/14/2021).

[Wox-1998] Woxenius, J.: Development of Small-Scale Intermodal Freight Transportation in a Systems Context. Dissertation. Göteborg: Chalmers University of Technology, 1998.

[Wox-2007] Woxenius, J.: Alternative Transport Network Designs and Their Implications for Intermodal Transhipment Technologies. In: European Transport, vol. 35 (2007), pp. 27–45.

[Wym-1993] Wymore, A. W.: Model-Based Systems Engineering. Systems Engineering Ser. Chapman and Hall/CRC, Boca Raton, 1993.

[Xin-2020] Xinhua: World's first 350 km/h freight train off assembly line. 2020. Url: http://www.xinhuanet.com/english/2020-12/23/c_139613705.htm (visited on 04/01/2021).

[Yee-2021] Yee, V.; Goodman, P. S.: Traffic Jam In Suez Canal As Huge Ship Runs Aground. In: The New York Times, Section A, (Mar. 25, 2021), p. 9.

List of Figures

A Specification of Groups of Goods

ID	Good Group	Example	Cluster	Transshipment Criteria: Load Carrier for Transshipment	Standardization Criteria: Load Carrier Standardization	Temerature Criteria: Temperature Control
1	Agricultural and forestry products; fish and fishery products	Fresh fruits and vegetables, Fresh eggs, plants and flowers	1	Pallet	yes	yes
2	Food and beverages	Milk and milk products (and processed)	1	Pallet	yes	yes
3	Textiles, clothing, leather, leather goods	Clothing	2	Pallet	yes	no
4	paper, paperboard and printings; wood and articles of wood, cork and plaiting materials; publishing and printing products, recorded sound, video and data media	Printing products	2	Pallet	yes	no
5	Chemical products and man-made fibers; rubber and plastic products; nuclear fuel	Pharmaceuticals	3	Boxes/ Roll Container	no	yes
6	Machinery and equipment; office machinery, computers and data processing equipment; electricity generating equipment	Electronic components; Medical, measurement, control and regulation products	4	Pallet	no	no
7	Automotive	Automotive and spare parts	4	Pallet	no	no
8	Furniture, jewelry, musical instruments, sports equipment, toys	Furniture	4	Pallet	no	no
9	Courier, Express and Parcel	Mail, parcels	5	Boxes/ Roll Container	no	no
10	Baggage transported separately from passengers, removal goods	Baggage	5	Boxes/ Roll Container	no	no
11	Consolidated cargo	Consolidated cargo	4	Pallet	no	no

Figure A-1: Specification and clustering of groups of goods based on transshipment, standardization and temperature criteria (DLR)

B SysML Reference Cards

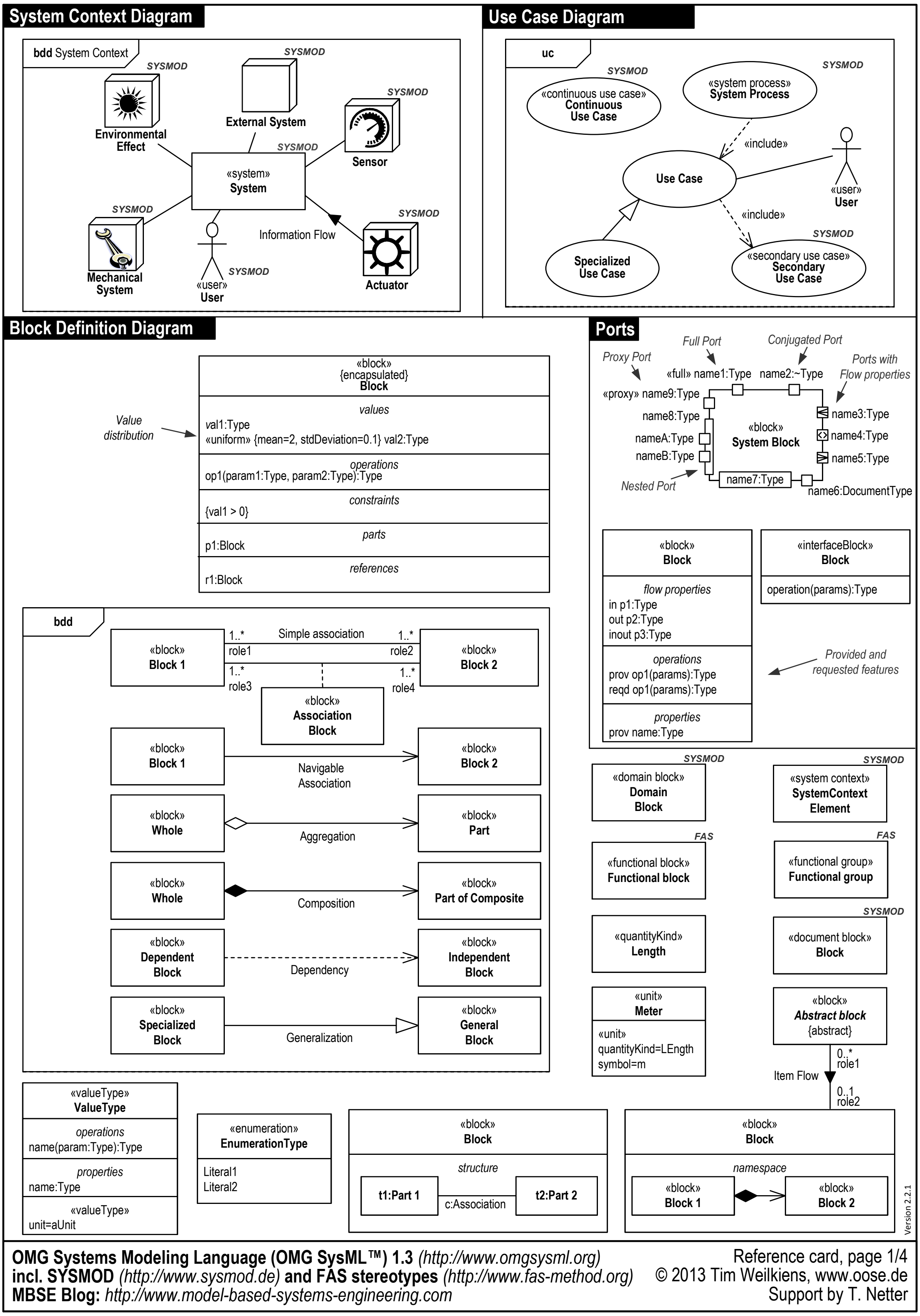


Figure B-1: SysML reference card (1/4) including SYSMOD profile [Wei-2021]

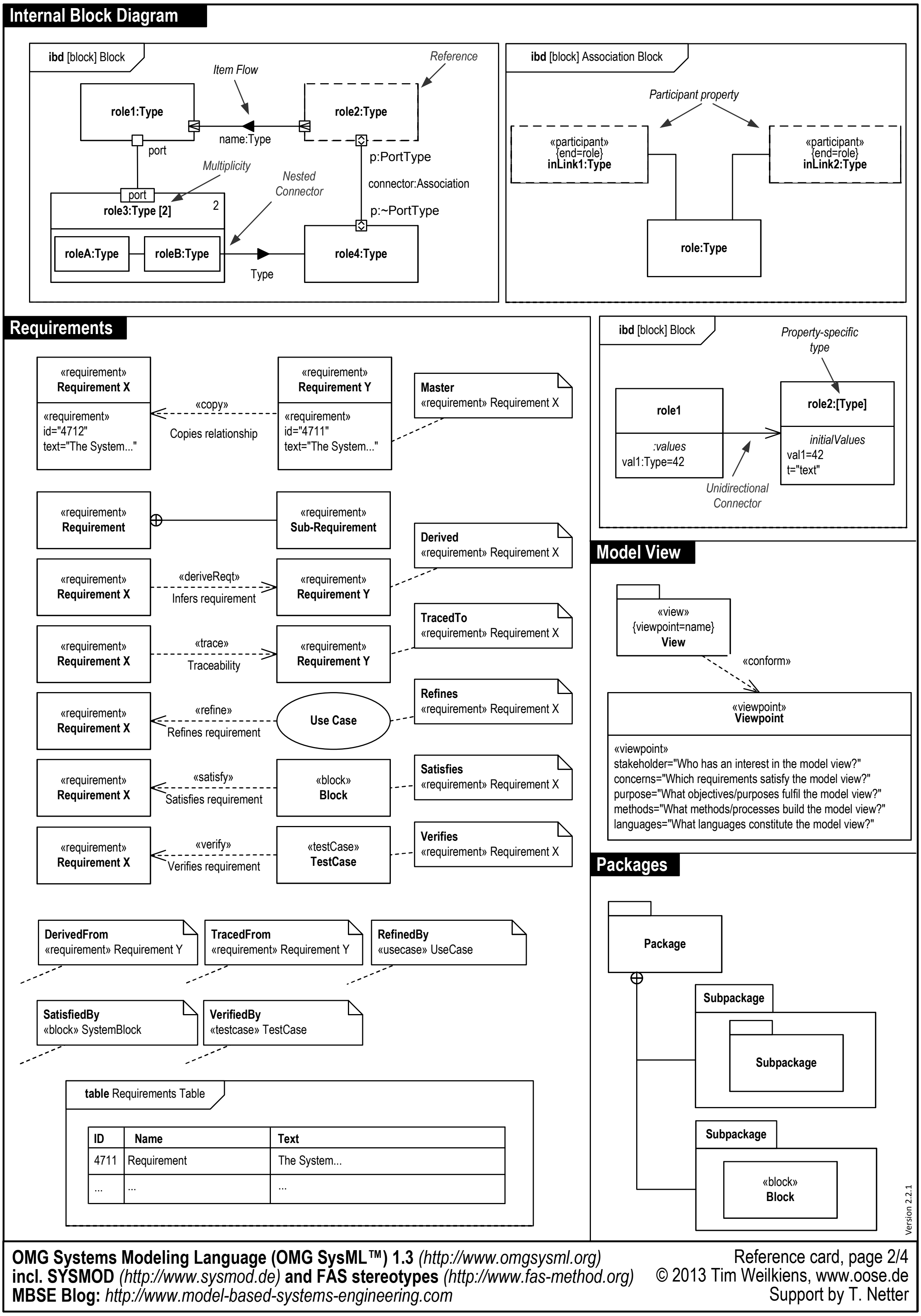


Figure B-2: SysML reference card (2/4) including SYSMOD profile [Wei-2021]

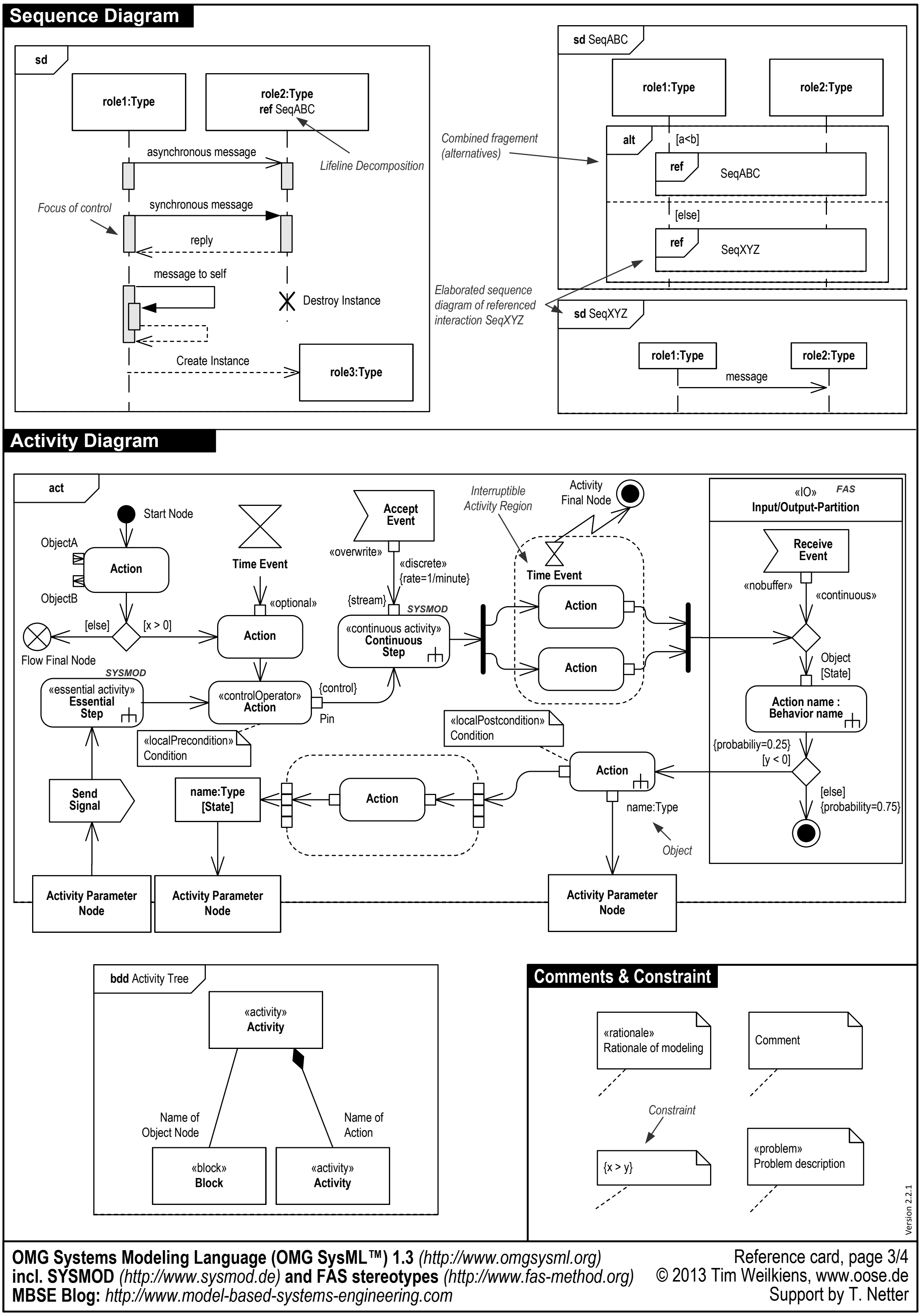


Figure B-3: SysML reference card (3/4) including SYSMOD profile [Wei-2021]

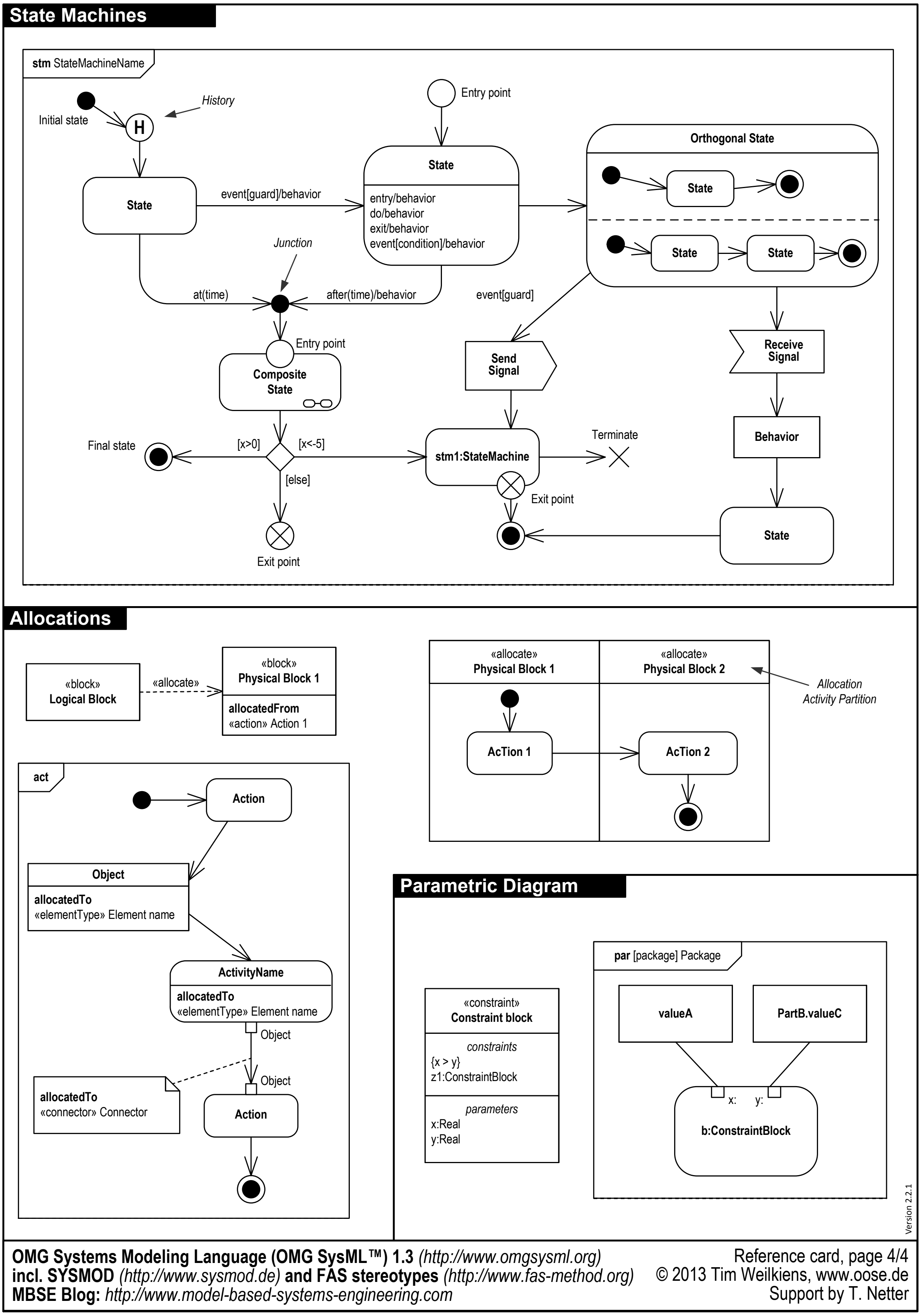


Figure B-4: SysML reference card (4/4) including SYSMOD profile [Wei-2021]

C Requirements of the Terminal Core

#	△ Name	Text
1	⊟ F 2 Functional Requirements	
2	⊟ F 2.1 Treatment of goods	The transshipment and storage of goods shall fulfill the requirements concerning handling, storing and monitoring depending on the type of good
3	F 2.1.1 Cool Chain	The cool chain of temperature-controlled goods shall not be broken during storage and transshipment
4	F 2.1.2 Storage Humidity	The humidity of storages shall be controlled if required by goods
5	F 2.1.3 Hygiene	The hygiene requirements depending on the goods shall be fulfilled by storage and handle equipmer and monitored
6	F 2.1.4 Dirt Protection	During storage and transshipment the goods shall be protected from dirt, dust, solar radiation and moisture
7	F 2.1.5 Fire protection	Storage and transshipment areas shall be fulfill the requirements concerning fire safety depending o the type of good
8	F 2.1.6 Theft-Protection	Storage and Transshipment areas shall not be entered by unauthorized persons
9	F 2.1.7 Electro Magnetic Tolerance	The requirements concerning Electro Magnetic Tolerance of good shall be met during transshipment and storing
10	F 2.1.8 Stacking load	The maximum permitted stacking load of load units shall not be exceeded during storing
11	F 2.1.9 Mechanical loads	The maximum permitted mechanical loads (acceleration, contact forces) of load units shall not be exceeded during transshipment and storing
12	F 2.1.10 Handling	The handling of goods shall be performed according to the permitted points of attack of external forces
13	F 2.4 Sort and Transship Load Units	Incoming load units shall be sorted and transshipped according to the loading of the units into the following mean of transport
14	F 2.5 Load Road Vehicles	Road vehicles of haulers shall be loaded with assigned load units
15	F 2.6 Unload Road Vehicles	Road Vehicles of Haulers shall be unloaded at the Terminal
16	F 2.7 Store Load Units	If required, load units shall be stored in the terminal before being transshipped
17	F 2.8 Monitor Freight Condition	The condition of load units shall be tracked during all processes in the terminal in order to prove damage (when, how, where) or not permitted handling including temperature and humidity, if requir
18	F 2.9 Provide Tracking Information	External systems shall be provided with Tracking information concerning the logistic transport chain and condition of loads units staying in the terminals
19	F 2.13 Unload Rail Wagons	Rail wagons shall be unloaded at the terminal
20	F 2.14 Load Rail Wagons	Rail wagons shall be unloaded at the terminal with assigned load units
21	F 2.17 Buffer Storage	The Terminals shall offer buffer storages at transshipment areas
22	F 2.18 Check Load Units	If required by transport standards or customs authorities, the content of load units need to be checked and controlled
23	F 2.22 Comunicate with WMS	Logistics Moduls shall be able to send and receive Information to/from WMS
24	⊟ U 3 Usability Requirements	
25	U 3.1 Load Units	The terminal shall be able to handle the load units of the goods
26	U 3.2 Degree of automation	Management, control and performance of loading, unloading, transshipment of load units as well as shunting procedures shall be performed applying a high but economically realistic degree of automation
27	⊟ P 4 Physical Requirements	
28	Ph 4.2 Load Unit Size	The load units transshipped in the terminal are -Europalette (L x W) 1.200m x 800m -Logistckbox / ULD: max. LD7 (L x W x H - 3.175m x 2.235mx 1.626m)
29	Ph 4.3 Order Picking Interface	The terminal's warehouse shall provide an interface to an external order picking system
30	Ph 4.4 Customs Interface	The terminal shall provide an interface to enable customs check by external control institutions.
31	⊟ P 5 Performance Requirements	
32	P 5.1 NGT Railcar Turnaround Time	NGT Rail Wagons shall be unloaded within 5 minutes
33	P 5.4 Energy Consumption	The consumption of energy of the terminal procedures shall be economically reasonable
34	P 5.5 Degree of Capacity Utilization	The Degree of Capacity Utilization of working funds of the terminals shall be as high as possible
35	⊟ R 6 Reliability Requirements	
36	R 6.1 Damaged Load	Damaged Load Units shall not affect the loading, unloading and transshipment procedures
37	R 6.3 Weather	All procedures at the terminals shall not be affected by the weather conditions

Figure C-1: Requirements of the terminal core (SysML view)

D Selection Criteria for Intralogistics Systems

Good Group: Printing products; Clothing;
Cluster: 2

Type	Conveying System	System is able to handle std. Euro-pallets	System is able to operate u. controlled Temperature	System in Evaluation?	Level of automation	Possibility of integration into automatic mat. flow sys.	Organization with data processing	Branching effort	Space requirements for transport routes	Storage and buffering capacity	Evaluation Result
	Good Groups Criteria: / Required by Good Groups:	yes	no								
	Selection Criteria: / Cluster:				Automation	Automation	Automation	Complexity	Complexity	Storage	
	Weighting:	100%	0%		16.7%	16.7%	16.7%	25%	25%	-	100%
Continuous Conveyor	Underfloor Drag Chain Conveyor	1	1	yes	2	1	2	2	3	3	69.4%
Continuous Conveyor	Roller Conveyor, Rubbed	1	1	yes	3	3	3	3	1	2	83.3%
Continuous Conveyor	Gravity Roller Conveyors	1	1	yes	2	2	1	2	1	3	52.8%
Continuous Conveyor	Wheel Conveyors	0	1	no	2	2	1	2	1	3	0.0%
Continuous Conveyor	Ball Transfer Table	0	1	no	2	1	1	1	1	2	0.0%
Continuous Conveyor	Slides	0	1	no	2	2	1	3	2	1	0.0%
Continuous Conveyor	Carrying Chain Conveyor	1	1	yes	3	3	3	2	1	2	75.0%
Continuous Conveyor	Belt and Link/ Apron Conveyor	1	1	yes	3	3	3	3	1	2	83.3%
Continuous Conveyor	Chain Conveyor System	0	1	no	3	3	3	2	1	3	0.0%
Continuous Conveyor	S- and C- Shaped Conveyor	1	1	yes	3	3	3	0	2	2	66.7%
Continuous Conveyor	Circular Conveyor System	1	1	yes	3	3	3	1	3	2	83.3%
Discontinuous Conveyor	Tractor	1	1	yes	1	1	2	3	3	2	72.2%
Discontinuous Conveyor	Pallet Truck	1	1	yes	1	1	1	3	3	1	66.7%
Discontinuous Conveyor	Order Picking Vehicles	1	1	yes	1	1	2	3	3	1	72.2%
Discontinuous Conveyor	Platform Stacking Truck	1	1	yes	1	1	2	3	3	1	72.2%
Discontinuous Conveyor	Fork Lift Truck	1	1	yes	1	1	2	3	3	1	72.2%
Discontinuous Conveyor	Order Picker Lift Truck	1	1	yes	1	2	2	2	1	1	52.8%
Discontinuous Conveyor	High Shelf Stacker	1	1	yes	1	1	2	3	3	1	72.2%
Discontinuous Conveyor	Stacker Crane (S/ RS)	1	1	yes	3	3	3	2	1	1	75.0%
Discontinuous Conveyor	Electric Conveying Track	1	1	yes	3	3	3	2	1	1	75.0%
Discontinuous Conveyor	Transfer Carriage	1	1	yes	3	3	3	3	1	1	83.3%
Discontinuous Conveyor	Automated Guided Vehicle	1	1	yes	3	3	3	3	3	1	100.0%
Discontinuous Conveyor	Lift/ Elevator	1	1	yes	2	1	2	0	2	1	44.4%
Discontinuous Conveyor	Channel-/ Shuttle Vehicle	1	1	yes	3	3	3	1	1	1	66.7%
Discontinuous Conveyor	Trolley Conveyor System	0	1	no	1	2	1	3	2	3	0.0%
Discontinuous Conveyor	Bridge/ Suspension Crane	0	1	yes	1	1	2	0	3	1	47.2%
Discontinuous Conveyor	Gantry Crane	0	1	no	1	1	2	0	2	1	0.0%

1 = yes; 0 = no;

3 = favorable; 2 = semi-favorable; 1 = unfavorable; 0 = N/ A;

Figure D-1: Evaluation matrix for conveying systems; groups of goods cluster 2 selected (own illustration based on Hompel et al. *[Hom-2018, pp. 240–242])*

Good Group: Printing products; Clothing;
Cluster: 2

Storage Type		Storage System	System is able to handle std. Euro-pallets	System is able to operate u. controlled Temperature	System in Evaluation?	Access Time	Accessability of every Load Unit	Utilization Degree	Space utilization	Prone to failure and accidents	Emergency Mode in case of Malfunction WH Equipment	Degree of Automation	Evaluation Result
		Cluster / Required by Good Groups:	yes	no		Speed		Capacity Effic.		Reliability		Autom.	
		Weighting:	100%	0%		12.5%	12.5%	12.5%	12.5%	12.5%	12.5%	25%	100%
Rack Storage (Dynamic)	Moving Racks, fixed LU	Mobile Shelving System	1	1	yes	3	3	3	3	2	1	2	79.2%
		Vertical Rotary Rack	0	1	no	3	3	2	2	2	2	3	0.0%
		Horizontal Rotary Rack	0	1	no	3	3	2	2	2	2	2	0.0%
	Fixed Racks, Moving LU	Live Storage Rack (Push) Disc. Conv. Gravity	1	1	yes	3	2	3	3	2	3	3	91.7%
		Live Storage Rack (Flow) Disc. Conv. Gravity	1	1	yes	3	2	3	3	2	3	3	91.7%
		Live Storage Rack (Flow) Cont. Conv. Drive	1	1	yes	3	2	3	3	2	2	3	87.5%
		Live Storage Rack (Push) Cont. Conv. Gravity	1	1	yes	3	2	3	3	1	3	3	87.5%
		Live Storage Rack (Flow) Cont. Conv. Gravity	1	1	yes	3	2	3	3	1	3	3	87.5%
Rack Storage (Static)	Row	Honeycomb Rack	0	0	no	3	3	2	2	3	3	3	0.0%
		Cantilever Rack	0	1	no	3	3	2	2	3	3	2	0.0%
		Shuttle System	1	1	yes	3	3	2	2	2	1	3	79.2%
		Container Rack	0	1	no	3	3	2	2	3	3	3	0.0%
		Pallet-/ High Bay Rack	1	1	yes	3	3	2	2	3	3	3	91.7%
		Drawer Shelf	0	0	no	3	3	2	2	2	3	2	0.0%
		Shelf Storage System	0	1	no	3	3	1	1	3	3	1	0.0%
	Block	Satellite Storage System	1	1	yes	3	2	3	3	3	1	3	87.5%
		Drive-in/ Drive-through Rack	1	1	yes	1	1	3	3	2	3	1	62.5%
Floor Storage	Row	Ground Bulk Storage (Line)	1	0	yes	2	1	2	2	2	3	1	58.3%
	Block	Ground Bulk Storage (Block)	1	0	yes	2	1	3	3	2	3	1	66.7%

1 = yes; 0 = no;
3 = favorable; 2 = semi-favorable; 1 = unfavorable;

Figure D-2: Evaluation matrix for storage systems; groups of goods cluster 2 selected (own illustration based on Hompel et al. *[Hom-2018, pp. 120–121])*

Declaration of Originality

Ich versichere hiermit, dass ich die von mir eingereichte Abschlussarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Munich, May 1, 2021