# Rail to Digital automated up to autonomous train operation

## D48.1 Autonomous Route Setting (AnRS) conceptual studies: use case list and concept definition

Due date of deliverable: 31/11/2024 (M24)
Actual submission date: 10/02/2025

Leader/Responsible of this Deliverable: Arne Lamm, DLR

Reviewed: **Yes**

| **Document status** | | |
|---|---|---|
| Revision | Date | Description |
| 01 | 02/11/2023 | Initial document |
| 02 | 30/11/2023 | Extend Chapter 1, 2 and 3 |
| 03 | 07/03/2024 | Extend Chapter 4 and 5 |
| 04 | 14/06/2024 | Rework Chapter 1-5 |
| 05 | 30/08/2024 | Extend Chapter 6 and Finalize Deliverable 48.1 |
| 06 | 11/10/2024 | Integrate internal Feedback |
| 07 | 25/11/2024 | Correction of SC comments |
| 08 | 10/02/2025 | Integrate external Feedback |



| **Project funded from the European Union's Horizon Europe research and innovation programme** | | |
|---|---|---|
| **Dissemination Level** | | |
| **PU** | Public | X |
| **SEN** | Sensitiv – limited under the conditions of the Grant Agreement | |

Start date: 01/12/2022 Duration: 42 months

## ACKNOWLEDGEMENTS

This project has received funding from the Europe's Rail Joint Undertaking (ERJU) under the Grant Agreement no. 101102001. The JU receives support from the European Union's Horizon Europe research and innovation programme and the Europe's Rail JU members other than the Union.

## REPORT CONTRIBUTORS

| Name | Company | Details of Contribution |
|---|---|---|
| Arne Lamm | DLR | Chapter 1, 2, 3, 4 5, and 6 |
| Francisco Parrilla Ayuso | INDRA | Chapter 4.3 and 4.4, Review Chapter 4 and 5 |
| Paula Díaz Serrano | INDRA | Chapter 4.3 and 4.4, Review Deliverable 48.1 |
| Hélène Arfaoui Kayanak | SNCF | Review Deliverable 48.1 |
| Marvin Damschen | RISE | Review Deliverable 48.1 |
| Paul Unterhuber | DLR | Review Deliverable 48.1 |

**Disclaimer**

## TABLE OF CONTENTS

# LIST OF FIGURES

# ABBREVIATIONS

## A

AI. *Artificial Intelligence*
AnRS. *Autonomous Route Setting*
ARCADIA. *Architecture Analysis & Design Integrated Approach*
ARS. *Automatic Route Setting*
ATO. *Automatic Train Operation*
ATP. *Automatic Train Protection*

## C

CBTC. *Communications-based Train Control*
CCS. *Control-Command and Signalling*
CSM RA. *Common Safety Methods for Risk Evaluation and Assessment*
CTC. *Centralized traffic control*

## D

DTO. *Driverless Train Operation*

## E

ERTMS. *European Rail Traffic Management System*
ETCS. *European Train Control System*
EU. *European Union*

## F

FFB. *dt. Funkfahrbetrieb*
FRA. *Federal Railroad Administration*
FRMCS. *Future Railway Mobile Communication System*

## G

GoA. *Grade of Automation*
GSM-R. *Global System for Mobile Communications-Railway*

## H

HMI. *Human-Machine Interface*

## I

IEC. *International Electrotechnical Commission*
IETF. *Internet Engineering Task Force*
IoT. *Internet of Things*
ISO. *International Organization for Standardization*

### M

MBSE. *model-based systems engineering*

### O

OCC. *operation control center*
ONRSR. *Office of the National Rail Safety Regulator*

### P

PTC. *Positive Train Control*

### R

RAMS. *Reliability, Availability, Maintainability, and Safety*
REM. *Réseau Express Métropolitain*
RTTP. *Real Timetable Plan*

### S

SAE. *Society of Automotive Engineers*
SCI-CC. *System Context Interface - Command and Control*
SIL. Safety Integrity Levels
SPAD. *Signal Passed at Danger*
SWOC. *Smart Wayside Object Controller*

### T

TCMS. *Train Control and Management System*
TD. *Technical Demonstrator*
TMS. *Traffic Management Systems*
TSI. *Technical Specifications for Interoperability*

### U

UIC. *engl. International Union of Railways*
UNECE. *United Nations Economic Commission for Europe*
UTO. Unattended Train Operation

# 1. EXECUTIVE SUMMARY

This concept study offers an in-depth exploration into the development of an advanced system called autonomous route setting (AnRS) for railway systems, with a specific focus on Grade of Automation 4 (GoA4). The study starts with an introduction to GoA4, detailing its role in achieving high levels of automation where human intervention is minimized, and transitions into a discussion on Automatic Route Setting (ARS), emphasizing its significance in optimizing railway operations.

The introduction also covers the latest technological advancements and future trends influencing this field. These include the integration of artificial intelligence (AI) and machine learning, which are novel decision-making processes and predictive maintenance. Sustainability efforts are highlighted, showcasing how modern systems aim to reduce environmental impact. The expansion of global railway networks is discussed, reflecting on how international trends shape local practices. Additionally, the study addresses the evolution of human-machine interactions, aiming to enhance user experience and operational efficiency, and emphasizes the importance of cybersecurity to protect complex automated systems from potential threats. The section concludes by discussing the benefits and challenges associated with these innovations, providing a balanced perspective on their practical implications.

In the "State of the Art" section, the document examines the functional requirements of GoA4, offering a thorough overview of current implementations and real-world applications. It explores regulatory considerations essential for compliance and presents an analysis of ARS functionality, including critical components such as the Smart Wayside Object Controller and Radio Driving Mode. This section provides a foundation for understanding how existing technologies are applied and regulated in practice.

The central focus of the study is the AnRS concept. This section initiates with the development of use cases, defining practical scenarios and applications. It then progresses to a detailed examination of requirements, including compliance with standards and guidelines such as railML, SCI-CC from EULYNX, IEC/ISO, and various regulatory frameworks. A comprehensive function specification follows, addressing both operational and system needs to ensure that the system meets its intended goals effectively.

The document further elaborates on the system architecture and conceptual integration, presenting a detailed design that aligns with existing infrastructure and interlocking interfaces. This ensures that the new system can be seamlessly integrated into current operations without causing disruptions. An operational concept analysis is included, featuring scenario planning and a demonstration case study. This case study illustrates the practical application of the AnRS system, providing a concrete example of how the proposed system operates in a real-world setting.

The study concludes by summarizing the key findings and emphasizing the transformative potential of autonomous route setting for the railway industry. The document highlights how this innovative approach can enhance safety, efficiency, and reliability in railway operations, setting a new benchmark for automation. By providing a detailed framework based on the ARCADIA method, the study aims to support future advancements and ensure the successful implementation of autonomous systems in the railway sector.

# 2. INTRODUCTION

Mobility is undergoing a radical change. While modes of transport are subject to constant and even increasing digitalization, the shortage of skilled workers is causing human errors that are increasing the potential for accidents. While the trend in the decentralized automotive sector is towards centralization[1], the centralized rail domain is moving towards decentralization [3–5]. This leads to conflicting objectives in control and safety technology and also in vehicle guidance. The railway industry has witnessed significant advancements in automation, leading to enhanced safety, efficiency, and capacity.
The Grades of Automation (GoA) levels, which will be described in more detail below (cf. Figure 2: Grades of automation [13]Figure 2), assume a high level of automation instead of autonomy[2] in the railway sector. Why the topic of autonomy in railway operations should also be considered is one aspect of this study. If the control and intervention options of a train and railway infrastructure are broken down to a sub-component, all actions on the infrastructure side end with the actuators, the railroad switch. This is just as essential for the strategic level (journey planning) as it is at the operational level (conflict resolution). Automating the railroad switch and also the higher-level decision-making can therefore be particularly useful for relieving the dispatcher/signaler. The following concept study is organized as follows in order to present the advantages and disadvantages in a comprehensible manner. Firstly, Chapter 2 introduces the topic by briefly outlining the current state of research and technology. This is followed by an explanation of the GoA level from an information technology perspective and an outline of automatic route setting, which serves as the basis and foundation for autonomous route setting (AnRS). These two will be discussed later in Chapter 3, followed by a description of the relevant technical components, which will be discussed in greater detail in this study. In order to be able to evaluate the proposed solution, the benefits and challenges will first be presented to highlight the research gap. Finally, in the introduction, we look at existing solutions from around the world (related work) from which we can learn for our concept study and bring this together with the regulations. Finally, we take a look into the future and show the trends that make the early development of an autonomous route setting indispensable. In Chapter 3, we will go into more detail on the GoA4 level and the functionality of the automatic route setting in order to systematically derive the requirements and use cases. In Chapter 4, use cases for the concept study is then derived based on real accidents and outlined as generalized use cases. The individual requirements from the partner views are then collected and listed. This is used to specify the functional description of the autonomous route setting. This functional view is translated into architectures in accordance with model-based system development, followed by a theoretical analysis of the concept in Chapter 5 and the conclusion in Chapter 6.

## 2.1. UNDERSTANDING GOA4

The planning tasks in the transport sector can be generalized across all domains and can be broken down into three levels [6, 7].

[1] Current highly automated driving solutions are for example limited in speed, as the perception or detection of the surroundings is limited by the range and therefore does not perceive enough to cover larger areas. This can be remedied by approaches in which the infrastructure assists in detecting the environment and therefore forwards situation images to several vehicles in order to be able to make safe decisions. See concepts such as managed automated driving (MAD) and collective perception services (CPS) in V2X [1, 2].

[2] Autonomous in general refers to the ability to act independently without external influence. This is not the case with the targeted systems in the transport sector, as the human fallback level is still being considered (cf. chapter 2.1.)

1. Strategic level
2. Operational level
3. Control level

The operational management tasks divide the second level (operational level) into the areas of short-term planning and operations command level. The control level is better known in the rail sector as the management or field level [6].

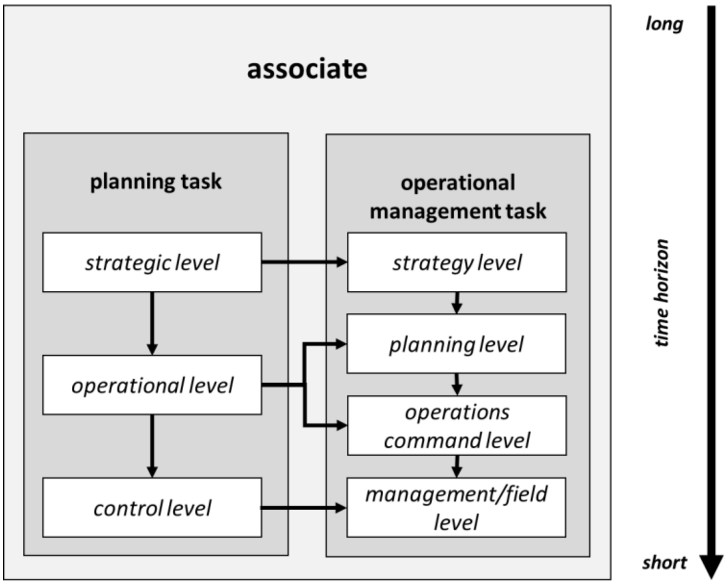


**Figure 1: Navigation Levels [6, 7]**

The tasks are not assigned to a specific person but could identify a role for each task, means it could be carried out in a distributed manner. Strategic planning is more of a long-term planning horizon. The tasks that fall under this are network expansion and the long-term timetable [6]. At the planning level, the essential tasks are the timetable, train composition and infrastructure planning. The operations command level are all activities around the disposition. These levels have a medium-term time horizon. The other task on the control level is the operations command level every activity around the train dispatching. The short planning horizon includes the tasks of the interlocking systems (both internal and external) [6]. The core objective of work package 48.1 is to develop an innovative solution for the railway sector that offers a practicable use case that falls under the term autonomy. One function that addresses all the levels described above is the route setting.

The term autonomy in the transport sector basically contains three criteria that need to be fulfilled [8, 9]. Firstly, there must only be a target for the system, without defining the process in detail and how the target is to be achieved because these systems will usually be applied in contexts with changing environments and fuzzy conditions (**adaptability**). Autonomous systems exhibit a higher degree of adaptability, adjusting their behavior based on real-time information and learning from experience. Automatic systems, on the other hand, follow predefined rules without the ability to adapt dynamically. Secondly, the possibility of external intervention that could override the system and its decision-making authority is not mandatory (**independence**). Autonomous systems are designed to operate independently for longer durations, requiring less frequent human intervention. Automatic systems typically need regular human oversight and intervention, especially in exceptional cases. That leads to the final criterion, the system must learn continuously and find a way to achieve its goals (**decision-making**). Autonomous systems have decision-making capabilities and can respond to unforeseen situations. Automatic systems rely on predetermined instructions and lack the ability to make complex decisions beyond their programmed parameters. With such systems, the question of security, reliability and trustworthiness inevitably arises. While both automatic and autonomous systems involve automation, the key distinction lies in the level of independence, adaptability, and decision-making capability. Automatic systems follow predefined rules and require more frequent human oversight, whereas autonomous systems operate with a greater degree of independence and can make decisions based on real-time data and learning algorithms [8–12].

The degrees of automation are already contrasted with types of operative processes. However, this only concerns the automation of train operation at an operational level. Train planning and signaling are not included in the GoA definition. In which control and responsibility is gradually transferred to automation. This requires a close exchange and coordination of information between train and infrastructure.

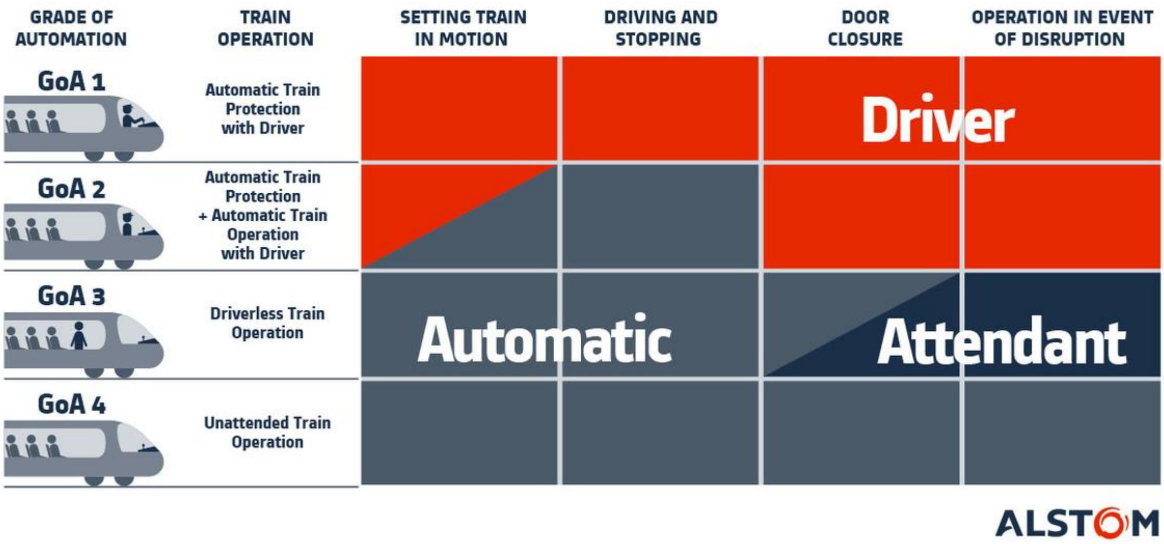


**Figure 2: Grades of automation [13]**

GoA 1 - Manual Operation with ATP: Moving up the automation scale, GoA 1 introduces Automatic Train Protection (ATP). The human operator remains in control of critical tasks such as door operation and emergency response, as well as setting train in motion the ATP system takes over assistance functions, such as speed monitoring. This level of automation enhances precision in speed control and station stops, contributing to overall operational efficiency.

GoA 2[3] - Automatic Train Operation (ATO): GoA 2 represents a significant step towards automation. Trains operating at this level can handle acceleration, braking, and station stops automatically. However, a human operator is still present on board, ready to intervene in case of unexpected events or emergencies. This collaborative approach combines the strengths of automation with the flexibility of human oversight.

GoA 3 - Driverless Train Operation (DTO): At GoA 3, the emphasis shifts towards fully automated train operation without the need for an onboard human operator. The DTO system manages all aspects of the train's journey, from route planning to emergency response.

[3] Additional types such as GoA2+ etc. are not included in this study.

Remote monitoring from a control center ensures that the entire rail network operates seamlessly, enhancing safety and efficiency.

GoA 4 - Unattended Train Operation (UTO): GoA 4 represents the highest level of automation, where trains operate without any onboard or remote human operators. The UTO system integrates advanced technologies for route optimization, obstacle detection, and emergency response. The entire rail network functions autonomously, providing a high level of efficiency and safety without direct human intervention.

The progression through these grades signifies a gradual shift from manual control to full automation in rail transportation. Each level introduces a higher degree of automation, leveraging technological advancements to enhance operational efficiency, safety, and capacity in the ever-evolving landscape of rail systems.

## 2.2. Automatic Route Setting

Automatic Route Setting (ARS) is a crucial component of modern rail systems, contributing to enhanced efficiency, safety, and overall automation [14]. This technology automates the process of determining and setting the optimal route for a train, considering factors such as the trains destination, schedule, and the current state of the rail network. The implementation of ARS plays a significant role in streamlining operations, reducing delays, and ensuring a more responsive and adaptive rail infrastructure [15]. It is a sophisticated technology integrated into rail signaling and control systems. Its primary objective is to automate the decision-making process related to route selection for trains, minimizing human intervention and optimizing the use of rail network resources [16]. Automatic route setting involves determining the optimal route for a train and can be implemented with any grade of automation, considering factors such as [14–16]:

Train schedule and destination

Track availability and occupancy

Speed limits

Track conditions and maintenance requirements

Safety regulations and interlocking with other routes

The aim is to ensure efficient and safe train movements while minimizing conflicts and delays. This automation is particularly crucial in complex rail networks where multiple trains operate simultaneously, and efficient routing is essential for maintaining schedules and preventing congestion. The Key components for ARS are [14–16]:

The **train scheduling system** to understand the planned routes and timetables for each train. It considers factors such as departure and arrival times, preferred routes, and any specific constraints related to train movements.

The **track and switch information** because ARS relies on real-time data from track sensors, switches, and other infrastructure elements. This information helps in assessing the availability of tracks, identifying potential conflicts, and determining the most suitable route for a given train.

An effective **communication system** to ensure seamless coordination between the ARS and various components of the rail network. This includes communication with train control systems, signaling equipment, and centralized control centers.

**Safety protocols** to prevent conflicting routes, avoid collisions, and adhere to signaling rules. It continuously evaluates the operational state of the rail network to dynamically adjust routes and prevent unsafe conditions.

ARS is often implemented in centralized control centers where operators monitor and manage the entire rail network. The system receives inputs from various sources, processes the information, and determines the most efficient routes for trains. ARS systems are designed to be interoperable with other rail control and signaling systems. They can integrate seamlessly with ATO, Train Control and Management Systems (TCMS), and other components of a modern rail infrastructure.

The future of Automatic Route Setting involves integration with established technologies such as the Internet of Things (IoT), 5G communication, and artificial intelligence. These technologies can further enhance the systems responsiveness and adaptability. As rail systems move towards higher levels of automation, the integration of ARS with ATO becomes increasingly important. This synergy allows for a more holistic and seamless approach to rail transportation. Additional, future developments in ARS may also focus on incorporating green and sustainable practices, optimizing routes to minimize energy consumption and environmental impact. By automating the decision-making process related to route selection, ARS contributes to operational efficiency, capacity utilization, and overall system reliability. As rail systems evolve to meet the demands of the future, the continued development and implementation of Automatic Route Setting will play a crucial role in shaping the next generation of intelligent and responsive rail transportation.

## 2.3. Technological Advancements and Future Trends

New technological innovations are driving advancements in the rail sector, particularly in enhancing safety. A key focus should be the implementation of an additional autonomous testing instance that can independently assess situations in real time. Ideally, this system would conduct decentralized, localized evaluations to verify overarching decisions and control command. Achieving this level of safety requires seamless integration of vast amounts of real-time data. Reliable infrastructure, effective communication, and detailed train information (currently underutilized in interactions with infrastructure) unlock new possibilities for innovative safety solutions. The following list provides a brief overview of the relevant technological components recommended for integrating autonomous[4] systems into existing rail infrastructure, with a particularly emphasis on optimizing safety in route setting.

1. **Realtime Traffic Management [17, 18]:**
    - Optimize the rail traffic in large railway networks equipped with mixed signaling systems.
    - Automatic, local traffic optimization and control by real-time train scheduling, routing and plan execution.
2. **Digital Signaling and Interlocking [17, 19–22]:**
    - Advanced digital signaling systems use real-time data to control routes dynamically.
    - Interlocking mechanisms prevent conflicting routes from being set, enhancing safety.
3. **Communication Networks [17, 23, 24]:**
    - High-speed communication networks enable real-time data exchange between trains, signals, and control centers.
    - 5G technology is increasingly being adopted to ensure low-latency communication.
4. **Predictive Maintenance [21, 25]:**
    - Machine learning and predictive analytics anticipate track and infrastructure maintenance needs, allowing proactive route adjustments.
    - Weather data and predictive modeling help optimize route planning.
5. **Onboard Sensors [17, 23, 25, 26]:**
    - Trains are equipped with onboard sensors, such as lidar and radar, to detect obstacles and monitor track conditions.
    - Data from these sensors informs route setting and collision avoidance.
6. **High Accuracy Train Positioning [24]:**
    - For better planning and scheduling of trains the information basis has to be exact.
    - Also, efficient usage and planning of railway infrastructure can be optimized.

Looking ahead, several trends are shaping the future of GoA4 and ARS and collectively aim to make railway operations safer, more efficient, environmentally friendly, and resilient to various challenges, ensuring a robust and sustainable future for the global rail industry.

[4] The following discussion focuses solely on autonomous systems whose properties have been predefined. The systems currently under consideration within the sector can, at best, be classified as highly automated rather than fully autonomous. These systems still necessitate human involvement, particularly through knowledge transfer either prior to or during operation. The subsequent use cases and activities will illustrate these trends in greater detail and highlight their significance.

### 2.3.1. Artificial Intelligence and Machine Learning

The following use cases and activities are used to explain the trends mentioned in more detail and outline their relevance.

*Predictive Maintenance [18, 21, 25]*: AI and machine learning algorithms can analyze vast amounts of data from train operations, maintenance logs, and environmental conditions to predict potential issues before they arise. This can help in scheduling maintenance more effectively, predicting breakdowns, and improving the overall reliability of the railway system.

*Decision-Making [17, 18, 26, 27]*: These technologies can assist in real-time decision-making by evaluating various scenarios and providing optimal solutions. For example, AI can help reroute trains efficiently in case of an unexpected disruption, minimizing delays and improving passenger experience.

### 2.3.2. Sustainability

The following use cases and activities will be used to explain the trends mentioned in more detail and outline their relevance.

*Energy Efficiency [19, 28, 29]*: There is a growing emphasis on reducing the energy consumption of railway systems. This can be achieved through smarter train operations, such as optimizing speed profiles and regenerative braking systems that feed energy back into the grid.

*Eco-Friendly Technologies [19, 28, 29]*: Innovations such as electrification of rail lines, use of renewable energy sources, and the development of hydrogen-powered trains contribute to making rail transport more sustainable. Automated systems can ensure these technologies are used to their fullest potential, further reducing the carbon footprint of railway operations.

### 2.3.3. Global Expansion

The following use cases and activities will be used to explain the trends mentioned in more detail and outline their relevance.

*Adoption of GoA4*: More countries and railway operators are recognizing the benefits of fully automated train operations. GoA4 involves no staff on the train, with all functions, including starting, stopping, and handling emergencies, controlled automatically. This level of automation can lead to increased efficiency, safety[5], and punctuality.

*Standardization and Interoperability*: As GoA4 systems are adopted globally, efforts are being made to standardize technologies, interfaces and protocols to ensure interoperability across different regions and networks, facilitating smoother international rail travel and freight transport.

[5] Safety in this context means reducing risks beyond regulatory compliance by minimizing human error and improving real-time hazard detection. The documented approach strengthens safety by leveraging automation for faster incident response, consistent adherence to protocols, and early fault detection. These enhancements contribute to a more reliable and resilient railway system, reducing the likelihood of accidents and operational disruptions.

### 2.3.4. Human-Machine Interaction

The following use cases and activities will be used to explain the trends mentioned in more detail and outline their relevance.

*Improved Interfaces*: Even with high levels of automation, human operators in operation control centers (OCC) play a crucial role in overseeing operations. Advances in user interfaces, such as more intuitive dashboards, augmented reality displays, and better alarm systems, can enhance the situational awareness and decision-making capabilities of these operators.

*Training and Simulation*: Advanced simulators and training programs can prepare human operators to manage automated systems effectively. These tools can provide realistic scenarios for operators to practice handling various situations, ensuring they are well-prepared for any eventuality.

### 2.3.5. Cybersecurity

The following use cases and activities will be used to explain the trends mentioned in more detail and outline their relevance.

*Threat Detection and Prevention*: As railways become more connected and reliant on digital systems, the risk of cyber-attacks increases. Enhanced cybersecurity measures are being implemented to detect potential threats early and prevent them from compromising the safety and efficiency of railway operations.

*Resilience and Recovery*: Strategies to ensure quick recovery from cyber incidents are also critical. This includes robust backup systems, incident response plans, and regular security audits to ensure that the railway systems can withstand and quickly recover from cyber threats.

## 2.4. Benefits and Challenges

Based on the previous survey, the benefits and challenges for autonomous systems in the rail domain can be summarized as follows [29, 30]:

### 2.4.1. Benefits

One of the primary benefits of highly automated systems in the rail domain is the enhancement of safety[6] through automated conflict avoidance. Automated systems can detect and respond to potential conflicts much more quickly and accurately than human operators, thereby reducing the likelihood of accidents. This increased safety is a significant advantage, as it can lead to fewer incidents and improved overall passenger and freight safety [31, 32].

Another significant benefit is the increase in network capacity. Automated systems can optimize train movements with precision, ensuring that trains run more efficiently and use the available tracks more effectively. This optimization can lead to an increase in the number of trains that can be run on the same tracks, enhancing the overall capacity of the rail network without the need for extensive physical infrastructure upgrades [31–34].

Improved punctuality and reduced delays are also notable benefits of automated systems in rail operations. By utilizing advanced algorithms and real-time data, these systems can predict and respond to potential delays, adjusting schedules and train movements to maintain a high level of punctuality. This improvement in service reliability is beneficial for both passengers and freight customers, leading to increased satisfaction and trust in rail services [31–34].

Cost savings are another advantage of applying automated systems. These systems can allocate resources more efficiently, such as optimizing fuel consumption and reducing wear and tear on equipment. Over time, these efficiencies can result in significant cost reductions for rail operators, making rail transport more economic viable [31–34].

Additionally, highly automated systems can contribute to a lower environmental impact. By optimizing speed profiles and reducing unnecessary stops and starts, trains can run more efficiently, resulting in lower fuel consumption and reduced emissions. This benefit is crucial in the context of increasing environmental concerns and the push for more sustainable transportation solutions [29].

### 2.4.2. Challenges

Despite the numerous benefits, there are also several challenges associated with the implementation of automated systems in the rail domain. One of the most significant challenges is the high implementation cost and the need for substantial infrastructure upgrades. Transitioning to automated systems requires significant investment in technology, infrastructure, and training, which can be a barrier for many rail operators.

Regulatory hurdles and safety certification also pose a challenge. The rail industry is highly regulated, and the introduction of new systems requires compliance with stringent safety standards. Obtaining the necessary certifications and approvals can be a complex and time-consuming process, potentially delaying the deployment of these systems.

[6] the risk of operator mistakes, misjudgments, or lapses in attention that lead to incidents.

Cybersecurity concerns are another major challenge. As automated systems rely heavily on connectivity and data exchange, they become potential targets for cyberattacks. Ensuring the security and resilience of these systems against such threats is critical to maintaining safety and trust.

Public acceptance and trust in fully automated systems are also a challenge. While the technology may be advanced, gaining the confidence of passengers, customers, and the society is essential. Overcoming scepticism and demonstrating the reliability and safety of autonomous systems will be a key challenge.

Finally, interoperability between different rail networks and technologies is an ongoing task. The rail industry often involves a mix of legacy systems and new technologies and ensuring that new systems can work seamlessly across various networks is complex. Achieving this interoperability is essential for the successful implementation and operation of autonomous rail systems.

### 2.4.3. Conclusion

In conclusion, while automated systems offer significant benefits such as enhanced safety, increased capacity, improved punctuality, cost savings, and reduced environmental impact, they also present challenges including high implementation costs, regulatory hurdles, cybersecurity concerns, public acceptance, and interoperability issues. Addressing these challenges is crucial for the successful integration of autonomous systems in the rail domain.

# 3. STATE OF THE ART

In the following Chapter the current state of the art about GoA4 in the railway domain is presented and discussed [30]. The main goal is to consider the existing regulations and real-world implementations, and based on that, describe the current challenges that must be adressed in the concept of autonomous route setting. To achieve this, the concept must be aligned with the functional requirements from the GoA4. The gaps and possible improvements need to be identified.

## 3.1. FUNCTIONAL REQUIREMENTS OF GOA4

The functional requirements of GoA4 in rail transportation refer to the specific capabilities and characteristics associated with ATO. While specific requirements may vary based on the system design and implementation, here we have derived three key functional requirements commonly associated with GoA4 [32]:

1. Full Automation of Train Operation:
   - **Description**: In GoA4, the train's operation is fully automated without the need for onboard or remote human operators. The entire journey, from departure to arrival, is managed by the onboard automation system.
   - **Functional Requirements**:
     - The system shall autonomously control train acceleration, deceleration, and speed
     - The system shall automate door operations, including opening and closing at stations
     - The system shall detect and respond to unexpected events or disruptions, such as obstacles on the track, without human intervention
2. Automatic Route Planning and Navigation:
   - **Description**: GoA4 systems require the capability to automatically plan and navigate the train's route, considering factors such as station stops, track conditions, and optimal paths.
   - **Functional Requirements**:
     - The system shall autonomously plan routes based on the train's schedule, station stops, and real-time network conditions
     - The system shall dynamically adapt to changes in track conditions, switch positions, or other unexpected events
     - The system shall integrate with signaling systems to ensure compliance with safety protocols and prevent conflicts with other trains
3. Advanced Sensor and Communication Systems:

- **Description**: GoA4 systems rely on advanced sensor technologies and robust communication systems to perceive the environment, detect obstacles, and communicate with the central control center.
- **Functional Requirements:**
  - The system shall implement sensor technologies, including LiDAR, radar, cameras, and other relevant sensors, for real-time environmental perception
  - The system shall continuously monitor track conditions, detecting and responding to obstacles, signals, and other critical elements
  - The system shall maintain reliable communication with the central control center for system status updates, reporting, and emergency response coordination

These functional requirements highlight the key aspects of fully automated, unattended train operation at GoA4. The seamless integration of these functionalities (cf. Figure 3) ensures a high level of autonomy, efficiency, and safety in the operation of rail transportation systems at this grade. It's important to note that specific implementations may include additional requirements based on the complexity of the rail network, regulatory standards, and safety considerations.

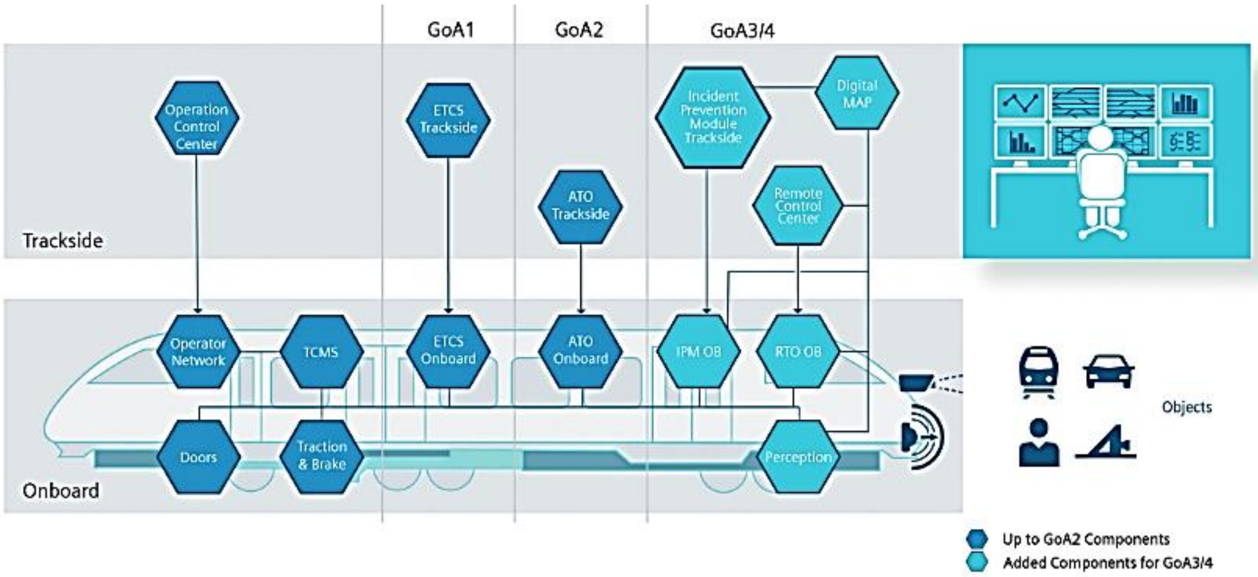


**Figure 3: Functions for GoA3 and GoA4 [32]**

## 3.2. Real-World Implementations

In the following, the current state of GoA4 systems will be examined and explained using real-life examples from around the world. The aim here is not to delve into the detailed workings of these systems, but to illustrate why fully autonomous rail operation is not yet widely adopted in the mainline rail sector and what the distinctive characteristics are, especially in comparison to the automotive industry.

**Europe**

- When we look at systems operating under GoA4, we see a variety of fully automated metro lines across different continents. For example, the Copenhagen Metro M3+M4 lines in Denmark are prime examples from Europe. These lines, which opened in 2019 and 2020 respectively, are driverless metro systems that operate under GoA4. The trains on these lines are fully automated, handling everything from station departures to arrivals, and they are equipped with advanced sensors and control systems that ensure precise and efficient operations.

**Asia**

- Moving to Asia, another noteworthy example is the Yurikamome Line in Tokyo, Japan. In operation since 1995, this line was one of the world's first fully automated train lines. The implementation of GoA4 on the Yurikamome Line allows for high-frequency and reliable service, as the automated systems can manage complex tasks such as route planning, obstacle detection, and emergency responses. The success of the Yurikamome Line underscores the long-term viability and benefits of GoA4 systems in urban transit environments.

**America**

- In America, the Réseau express métropolitain (REM) in Canada can be mentioned. This new system is the most significant public transit project in Québec since the launch of the Montréal Metro in 1966. While it will primarily serve suburban areas, it is expected to enhance public transit service for over four million residents in the Montréal region. Operating under GoA4, these lines are fully automated. The project is not yet completed and has been extended multiple times.

**Australia**

- In Australia, the Sydney Metro, which began operations in 2019, features fully automated trains under GoA4. This system, the first of its kind in Australia, serves as a showcase of modern rail technology in the Oceania region, offering high-frequency, reliable service across the Sydney metropolitan area.

The solutions mentioned are predominantly metro systems. For reasons, there is currently no operation at the GoA4 level on mainline railroads, which typically serve longer distances and more complex routes. It is difficult to secure the track from one station to another, like it is possible for metro stations. The complexity are several steps higher. The primary solutions presented here around the world include Alstom Urbalis 400, Siemens Trainguard MT CBTC, Hitachi Rail STS CBTC, Bombardier CITYFLO series, and Thales' SelTrac.

These solutions are often built upon a communications-based train control (CBTC) system. CBTC involves intensive communication between the train and the track infrastructure throughout the entire journey. The trackside infrastructure sends commands for speed limits or stopping points to the vehicle, ensuring safe and efficient operations. In an emergency, operators can always intervene from an OCC, providing an additional layer of safety. However, this extensive route infrastructure is particularly costly for branch lines, which often makes it financially impractical to equip these lines with the necessary instrumentation.

In comparison to the automotive sector, SAE Level 5, which is the equivalent of GoA4, does not rely on such infrastructure-based safety measures, yet. But no certified SAE Level 5 system exists at the moment. Instead, the decision-making power and management of critical situations are entirely entrusted to the vehicle itself. This fundamental difference highlights why fully autonomous operation is more challenging and less widespread in the rail sector, where the operational environment involves unique constraints and complexities related to infrastructure and safety protocols.

## 3.3. Regulatory Considerations

The introduction of GoA4 systems requires a comprehensive regulatory framework:

- Standardization of safety requirements and certification processes.
- Collaboration between railway authorities and regulatory bodies.
- Compliance with international railway safety standards.

Regulations need to keep pace with technological advancements while ensuring safety remains paramount. Following regulations do exist and relates to and/or specify rules and regulations for safe automated train systems.

1. **European Union (EU) - Technical Specifications for Interoperability (TSIs):** The EU has established TSIs that set out the essential requirements for the design and construction of railway systems. For GoA4 systems, compliance with TSIs related to control command and signalling subsystems, as well as interoperability constituents, is essential.
2. **International Union of Railways (UIC) - Code 505:** The UIC has developed Code 505, which addresses the functional and technical requirements for fully automated operation of trains. It covers aspects such as train integrity, obstacle detection, communication, and interfaces between the train and the infrastructure.
3. **United States - Federal Railroad Administration (FRA):** In the U.S., the FRA oversees railroad safety. Regulations related to Positive Train Control (PTC) are particularly relevant to automated train systems. PTC systems aim to prevent train-to-train collisions, overspeed derailments, and certain human-factor-caused accidents.
4. **Japan - Railway Business Act:** Japan has its own set of regulations governing the railway industry. The Railway Business Act outlines safety standards and requirements for railway operations, including those involving automated systems.
5. **Australia - Office of the National Rail Safety Regulator (ONRSR):** In Australia, the ONRSR oversees rail safety regulation. Regulations related to train control systems and safety management systems are critical for ensuring the safe deployment of GoA4 technology.
6. **International: United Nations Economic Commission for Europe (UNECE) - Rail Traffic Management System (ERTMS):** The UNECE has been involved in the development of the ERTMS, which includes specifications for train control and command systems. ERTMS aims to harmonize train control systems across Europe and beyond. The successful implementation of ATO GoA2 relies on secure and standardized communication, which benefits from IETF's protocols. Together, IETF's internet standards and ERA's rail safety regulations ensure that automated rail systems like ATO can operate securely and efficiently across Europe.

It's important to note that the regulatory landscape for GoA4 systems is dynamic, and regulations are subject to updates and revisions. Additionally, different countries may adopt standards and regulations specific to their needs and contexts. Regulations for GoA4 systems in the railway industry typically address various key requirements to ensure the safe and effective operation of fully automated trains. While specific requirements may vary by region and regulatory body, common themes include:

1. **Safety Assurance:** Ensuring the safety of passengers, crew, and the public is mandatory. Regulations require safety assessments, hazard analyses, and risk mitigation strategies to minimize the likelihood of accidents and ensure a high level of safety in highly automated train operations.
2. **Train Integrity:** Regulations often specify requirements for maintaining the integrity of the train consist. This includes ensuring that all wagons of a train remain connected and communicate effectively, preventing issues such as unintended decoupling.
3. **Obstacle Detection and Collision Avoidance:** automated trains must be equipped with advanced sensor systems to detect obstacles on the tracks and implement collision avoidance measures. Regulations mandate the use of technologies such as radar, lidar, and cameras to provide a comprehensive view of the railway environment.
4. **Communication Systems:** Effective communication between trains and with the railway infrastructure is crucial for safe and coordinated operations. Regulations stipulate the standards for communication protocols, ensuring that data exchange between trains and infrastructure components is reliable and secure.
5. **Interoperability:** Standards for interoperability are essential, especially in regions with multiple railway operators or cross-border operations. Regulations aim to establish common standards to facilitate the interoperability of autonomous trains, enabling them to operate seamlessly across different networks.
6. **Cybersecurity:** Given the increased reliance on digital systems, regulations address cybersecurity concerns. Requirements focus on protecting train control and communication systems from unauthorized access, cyberattacks, and other potential security threats.
7. **Human-Machine Interface (HMI):** Regulations often include guidelines for the design of human-machine interfaces to ensure that operators and passengers can interact with the autonomous system effectively. This includes clear displays, user-friendly controls, and appropriate communication of system status.
8. **Emergency Response and Fail-Safe Mechanisms:** Regulations require the implementation of robust emergency response procedures and fail-safe mechanisms. Autonomous trains must be capable of handling unforeseen situations, such as system failures or unexpected obstacles, in a manner that minimizes risks and ensures passenger safety.
9. **Testing and Certification:** Before deployment, regulations typically mandate comprehensive testing and certification processes. This includes both laboratory and real-world testing to validate the performance, safety, and reliability of the autonomous train systems.
10. **Data Recording and Analysis:** Regulations often require the implementation of data recording systems to capture information related to train operations, incidents, and system performance. This data is crucial for post-incident analysis, system improvement, and regulatory oversight.

These key requirements collectively aim to establish a regulatory framework that fosters the safe, reliable, and interoperable deployment of Grade of Automation Level 4 systems in railway operations. It's essential for stakeholders, including railway operators, manufacturers, and regulatory bodies, to collaborate closely to meet these requirements and advance the responsible adoption of autonomous train technology.

## 3.4. FUNCTIONALITY OF AUTOMATIC ROUTE SETTINGS

ARS in the rail domain is a crucial aspect of railway operations, ensuring the safe and efficient movement of trains. This functionality is integrated into the interlocking systems, which are safety-critical control systems. These systems prevent conflicting routes and ensure that only safe routes are set for trains.

At the core of ARS is the utilization of route optimization algorithms. These algorithms consider factors such as train schedules, track availability, and potential conflicts to determine the most efficient and safe routes for trains. The process involves continuous monitoring of data (Figure 4).

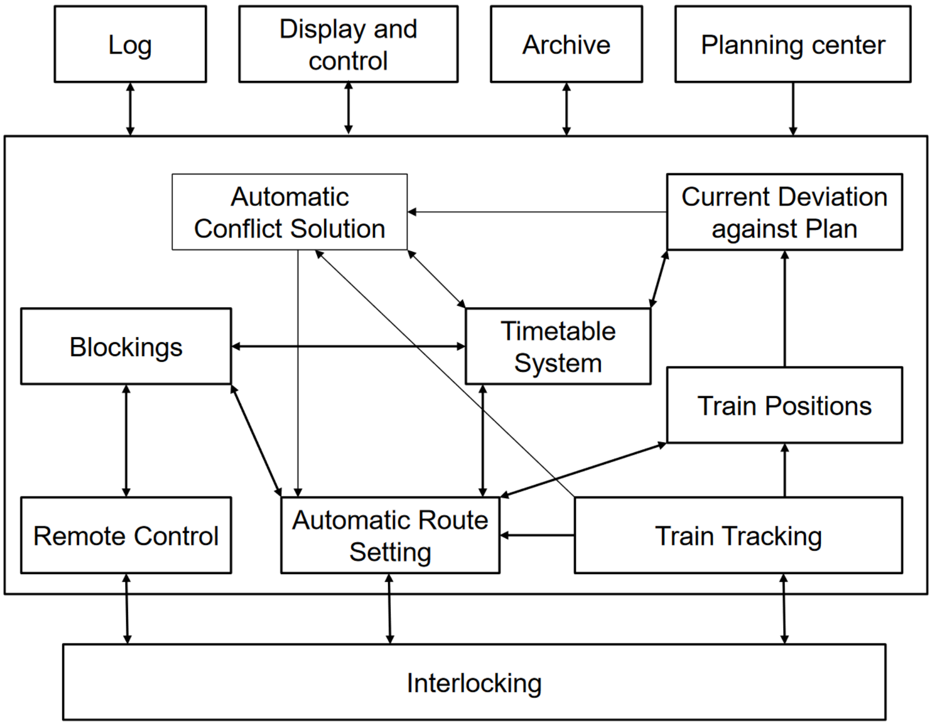


**Figure 4: Traffic Management System Architecture with ARS [35]**

Rail networks often have centralized control centers where operators or computer systems manage train movements. ARS is a key feature of these control systems, communicating with trackside equipment such as signals and switches.

Integration with train control systems is necessary, allowing for seamless coordination between the infrastructure and trains. The system continuously monitors train positions, adjusting routes to accommodate changes in schedules, unexpected delays, or other operational factors.

Security measures are incorporated to prevent unauthorized access or tampering, emphasizing the importance of maintaining the integrity and safety of rail operations.

Redundancy is also a key consideration, ensuring reliability by implementing backup mechanisms in case of system failures.

In summary, automatic route setting in the rail domain involves a sophisticated process that combines safety-critical control systems, real-time data processing, and seamless integration with various elements of the rail infrastructure. The goal is to optimize railway operations through the improvements of train movements guarantying the safety of themselves.

### 3.4.1. Smart Wayside Object Controller

Next to the ARS approach we have reviewed current activities and novel concepts around interlocking and route setting. In the scope of the X2Rail-4 project for the Technical Demonstrator TD2.10 Smart Radio- Connected all-in-all way side objects, also known as Smart Wayside Object Controller (SWOC) was introduced. The objective of TD2.10 was to demonstrate a decentralized approach to rail automation through the use of object controllers. This approach is designed to be scalable, making it suitable for high-performance lines as well as regional and freight applications. Even though modern signalling systems have significantly reduced the amount of trackside equipment, this solution remains relevant because interfaces to points, level crossings, and other essential components will still be required. The motivation for that approach was, that at the moment, providing power and data cabling to remote trackside objects is costly. Cables are vulnerable to theft, changes in track layouts are complex and expensive, and the cables limit the distance between trackside objects and signalling equipment. Reducing the amount of cabling will lead to significant reductions in lifecycle costs. Within the scope of TD2.10, system requirements and architecture for SWOC have been developed, and these will be further refined and tested in the demonstrator [36–38].

### 3.4.2. Radio Driving Mode[7]

Another relevant approach that is quite similar to the AnRS approach because of the decentralization of railway functionality is the radio driving mode (ger. Funkfahrbetrieb; FFB). The core idea was to shift the safety logic from centralized signal boxes to the vehicles themselves. The vehicle should autonomously initiate and monitor its route settings via a radio interface. Additionally, level crossings should also be activated through a radio connection. Moving the safety logic into the vehicle requires that the vehicle knows its position within the network. To achieve this, passive beacons laid in the track are used for localization, and the data is then compared with a track map stored on the vehicle.

A route is requested by the vehicle via radio from a central control. The control center only grants permission if the route is clear. Similarly, access to the control of switches and level crossings is only granted by the central control when conditions allow. That means, that the monitoring of train integrity must be handled on the vehicle side. The driver's cab display developed for radio-based train operation was, by the end of the 1990s, almost identical to the Driver Machine Interface of ETCS, with only a few insignificant differences. The

[7] National funded project: further information can be found here: M. Schaefer and W. Pauli, "Funkfahrbetrieb-Erstanwendung für die DB AG im Lautertal," *Signal+ Draht Heft März,* 2000.

implementation was expected to make the operation of secondary railways more economical by eliminating the need for signal boxes, signals, and cabling. However, on the pilot route "Haller Willem," no approval had been granted by 2004. The anticipated cost savings were not achieved, primarily because the costs of the radio system were underestimated. Another challenge is the transfer of safety technology costs from the infrastructure to the vehicle, especially given the separation between railway infrastructure companies and railway operating companies since the railway reform [39].

### 3.4.3. Summary

It can be summarized that the Radio Driving Mode and the SWOC have limitations compared to fully autonomous systems. Radio driving modes still rely on human intervention, which introduces variability in decision-making and responsiveness. Smart wayside object controllers, on the other hand, are limited to localized control and cannot optimize the entire network's routing, failing to account for real-time factors across a broader area.

# 4. AUTONOMOUS ROUTE SETTING CONCEPT

In order to better understand and categorize the concept of autonomy, the use case of anomaly detection in safety-critical infrastructure such as server environments will be described first. Autonomous anomaly detection in server environments prevents hacker attacks by continuously monitoring and analysing various data sources like network traffic, system logs, application logs, and user behaviour. The system begins by collecting this data and using it to establish a baseline of normal activity through historical analysis. Once the baseline is set, the system continuously monitors real-time data, comparing current activities against the established norms. It uses statistical methods and machine learning algorithms to detect any deviations from the normal behaviour. When an anomaly is detected, such as multiple failed login attempts or unexpected network traffic patterns, the system flags it and alerts administrators [12].

In response to detected threats, the system can automatically take measures to mitigate risks. This might include blocking suspicious IP addresses or isolating compromised servers to prevent further damage. The system also continuously improves by updating its models with new data and feedback, enhancing its ability to detect and respond to emerging threats over time. This proactive approach ensures that potential security threats are identified and addressed swiftly, reducing the likelihood of successful hacker attacks.

In summary, it can be said that no precise goal is set, but the system for recognizing unwanted access learns what it should look for and makes decisions without intervention.

Several parallels can already be drawn here between the railroad domain and computer networks. The traffic is systematically guided and routed. There are nodes at which it is often necessary to act and react reactively, but there are also descriptions for finding the regular route to the destination (server or station). So why not take the approaches from network technology and adapt them for rail operations?

The proposed concept therefore addresses the core requirements already derived for the benefits that are already to be considered and makes it possible to be seamlessly integrated into the existing infrastructure or, more precisely, the interlocking through well-defined interfaces (cf. Figure 5). The railML interlocking subschema contains definitions of data describing railway signalling and the use of interlocking systems. Interlockings in strict sense are control systems using movable elements, signals, detectors, and other components in combinations and sequences that hinder collision and derailment of trains. [8]

[8] https://wiki3.railml.org/wiki/User:RailML_Coord_Documentation/Tutorial/Interlocking_schema_of_railML_3 (last checked 11/10/2024)

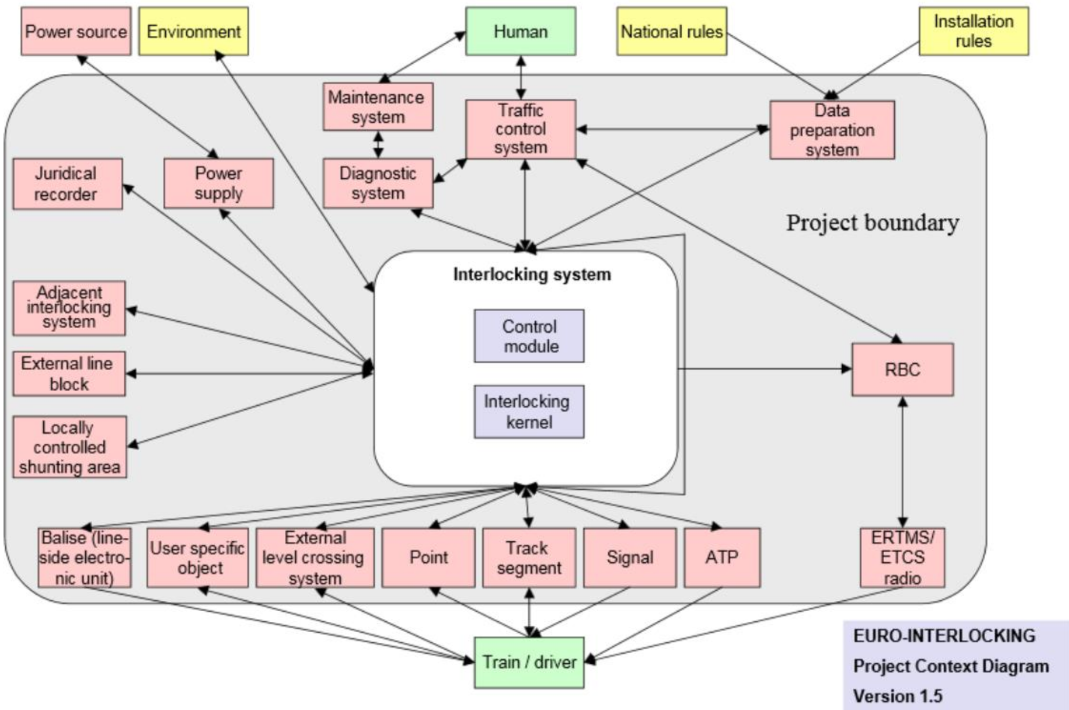


**Figure 5: RailML interlocking architecture**

We propose a distributed system[9] for route setting because it can be more useful than a centralized one for several reasons. In a centralized system, all processing and decision-making are done at a single point or within a single authority. This means that if that central point fails or is compromised, the entire system can lead to accidents or incidents, leading to a single point of failure (human factor).

In contrast, a distributed or decentralized system spreads out the processing, data storage, and decision-making across multiple nodes or locations. In a distributed system, different parts of the system can operate independently, communicating and coordinating with each other just when needed. If one part of the system fails, the other parts can continue functioning, which makes the system more resilient and reliable. Decentralization also enhances security because there isn't a single point that attackers can target. Furthermore, distributed systems can handle growth more efficiently because the workload is shared across multiple nodes, preventing bottlenecks.

Decentralization also empowers individual nodes or participants by giving them more control over their own data and processes, reducing dependency on a central authority. This can be particularly beneficial in situations where trust, autonomy, or data privacy is important. In this context we talk about edge computing. Edge computing means processing data closer to where it is generated (at the "edge" of the network), rather than relying solely on a centralized data center. This approach aligns naturally with the principles of distributed or decentralized systems.

In edge computing, data processing and decision-making occur at various distributed nodes, which might be devices, local servers, or other points near the data source. This reduces the

[9] In a way of a self-organized system-of-systems

need to send all data back to a central location for processing, which in turn reduces latency, improves response times, and lowers the amount of bandwidth needed.

It enhances efficiency by allowing data to be processed locally, meaning that critical decisions can be made quickly. This is particularly important in time-sensitive applications, like highly automated trains. Processing data locally, edge computing also enhances privacy and security. Sensitive data doesn't have to travel across the network to a central server, reducing the risk of interception or exposure. Each edge device or node can manage its own data, making the overall system more secure and less vulnerable to attacks. And edge computing can lead to cost savings. By reducing the need to transmit large volumes of data to a central location, it lowers the bandwidth and cloud storage requirements. This is especially beneficial as the amount of data generated by IoT devices and other sources continues to grow. Moreover, edge computing increases the system's resilience. In a distributed or decentralized system, if one node fails, the others can still function, ensuring that the system continues to operate effectively. This resilience is crucial in environments where uptime and reliability are critical.

## 4.1. Use Cases

To identify the initial need for an Autonomous Route Setting System we analysed the ERail Database[10] and especially accidents and incidents where the “Occurrence description” identified the influence of the ‘signalling’ and the ‘switches’ in the accident occurrence process. The data set contains detailed records of 3,614 railway incidents reported across various countries (2002 - 2023). The data includes key details such as the type of occurrence (e.g., train derailments, level crossing accidents, train collisions), the country where the incident occurred, the reporting body, and the date and time of each incident. The most common types of occurrences are train derailments and level crossing accidents, with Germany, France, and Italy being the most frequently reported countries.

The data also shows trends over time, with the number of incidents peaking in certain years, notably 2017. It has to be mentioned, that some countries have dominant reporting bodies that handle a large volume of investigations. For example, the Federal Bureau of Railway Accidents Investigation in Germany is responsible for 522 reports, indicating a significant concentration of data from Germany. This could suggest different kind of reporting mechanisms between countries. Nevertheless, There’s a noticeable decline in reported occurrences in recent years, particularly in 2023. This suggests a need to address potential improvements in railway safety, underreporting, and delays in data entry over recent years.

[10] https://data.europa.eu/data/datasets?locale=en (last checked 11/10/2024)


| Number of reports | Country |
|---|---|
| **512** | Czech Republic |
| **462** | Hungary |
| **446** | Romania |
| **370** | United Kingdom |
| **345** | Spain |
| **231** | Germany |
| **152** | Austria |
| **149** | Norway |
| **104** | France |
| **98** | Finland |
| **85** | Italy |
| **84** | Denmark |
| **81** | Portugal |
| **70** | Croatia |
| **70** | Poland |
| **58** | Bulgaria |
| **58** | Sweden |
| **57** | Belgium |
| **52** | Ireland |
| **34** | Slovak Republic |
| **21** | Slovenia |
| **18** | Estonia |
| **18** | Greece |
| **13** | The Netherlands |
| **11** | Latvia |
| **7** | Switzerland |
| **3** | Lithuania |
| **2** | Luxembourg |
| **2** | Slovakia |

**Figure 6: Overview of European Train Accidents**

From 3,614 accidents between 2002 and 2023 (stand 01.03.2023; open and closed reports), we identified 80 accidents with a direct connection to the operation of the interlocking. Further we analysed the accident reports that are allocated to the “Occurrence Type” ‘Trains collision’, ‘Train derailment’ or ‘Trains collision near miss’. The data set in general contains 356 accidents of type ‘train collision’, 1,157 accidents of type ‘Train derailment’ and 25 accidents of type ‘near miss’. Five near misses and ten collisions have a relation to the interlocking. For the concept study we will focus on the analysis of the near misses and collisions, because the first investigation showed, that these ‘Occurrence Types’ in general can be referred to as human errors.

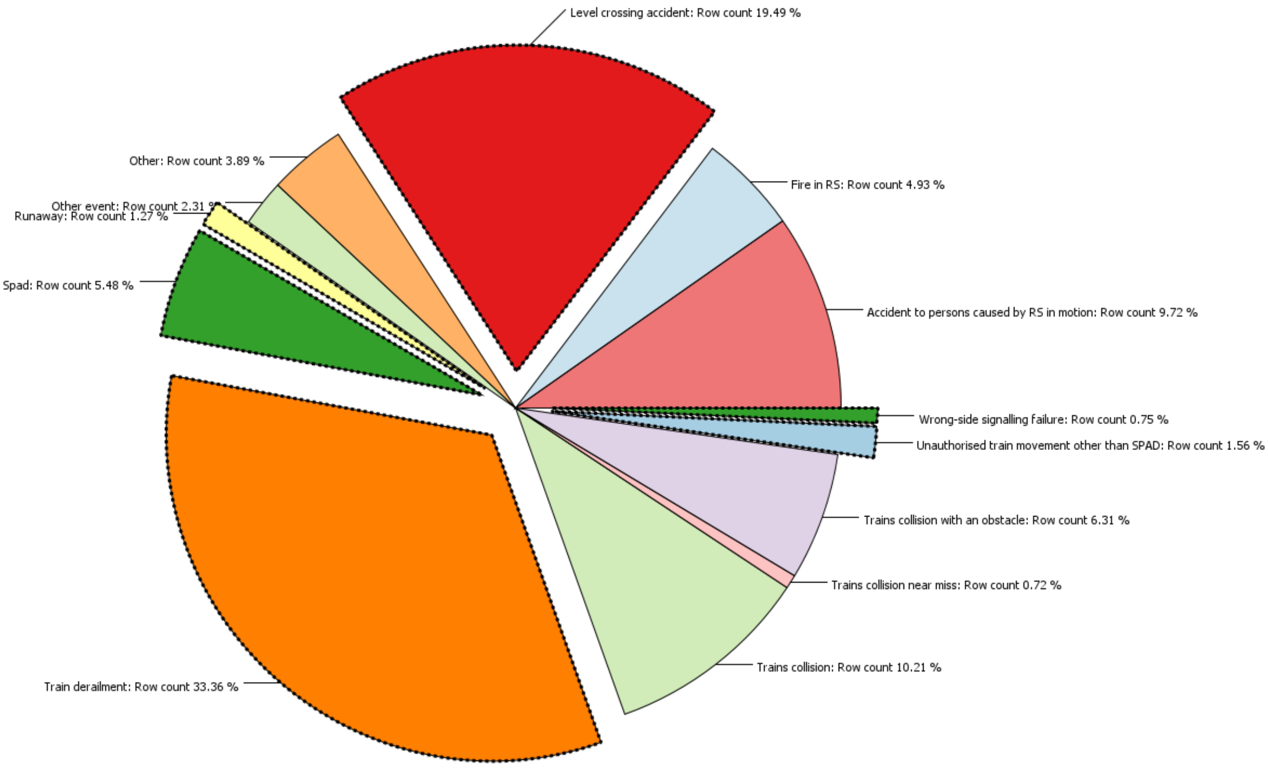


**Figure 7: Type of occurrence**

Figure 7 illustrates the variance observed among different types of accidents within the existing dataset, highlighting the distribution and frequency of each category, as well as the overall trends and patterns that emerge from the analysis. While train derailments and level crossing accidents dominate the dataset, there are over 30 distinct types of incidents recorded. While train derailments and level crossing accidents consistently account for a large portion of incidents, other types such as Signal Passed at Danger (SPAD) and train collisions have varied in frequency over the years. This indicates evolving challenges in railway safety. This diversity highlights the range of risks and challenges faced by railway systems, from operational events like SPAD to more specific incidents like broken wheels or electric shocks. The spread of incidents across countries shows that railway safety issues are not confined to a specific region but are a widespread concern across Europe, with even smaller countries like Luxembourg and Lithuania contributing to the dataset.

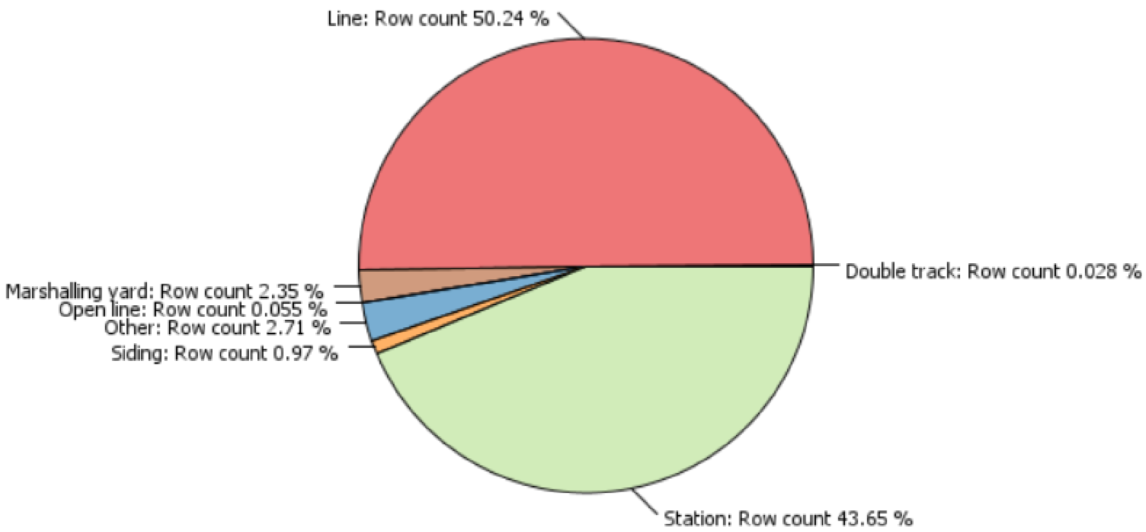


**Figure 8: Type of location**

Figure 8 shows statistic about where the accidents occurred. To generate a better understanding how we derived the use cases for the Autonomous Route Setting in the following 3 of these 15 accident reports will be described in detail which are representative for the other accident reports in the same category.

### 4.1.1. Accident report – SE-511

| *Report Type* | *Investigation Status* | *ERAIL Occurrence* | *Title* | *Reporting Body* | *Date of occurrence* |
|---|---|---|---|---|---|
| **Final report** | **Closed** | **SE-511** | **Trains collision near miss, 2008-06-17, Open-line site Klockarbacken on the main line between Vännäs and Umeå. (Sweden)** | **Swedish Accident Investigation Board** | **2008-06-17** |
| "Tuesday, 17 June 2008 at. 07:50 an incident occurred to a collision between a passenger train and wagon 7081 Competition 76910 on the route between Umeå and Brännland. Distance traffic was headed by a remote dispatcher on traffic control in Boden. That day, the dispatcher started his work with taking over from the dispatcher that had monitored the line. After taking over, he discovered that the trains real positions were not entered on the graphical display, so he started his work shift with entering the real postions of the trains. The picture over the track layout over Vännäs was moved, and the dispatcher had to find | | | | | |

new reference points in order to get it right. According to the dispatcher, the lines monitored had a lot of traffic and switching. After a while the phonesupport fpr the CTC went out of service and the dispatcher eperienced the situation as quite stressful. At 07:16m the CTC dispatcher was contacted by the supervisor for movement 76910, informing that he was ready to drive to Klockarbäcken. the dispatcher gave a start permission. After having a green light, the train was driven to Klockarbäcken. The supervisor opened the switch to the siding and shunted wagons there. at 07:47, the dispatcher was contacted by the driver on passenger train 7081 positioned at signal 2/5 in Brännland. The driver informed the dispatcher that he had a red signal. The dispatchers screen showed that the signals showed "stop". When he looked at the graphical display, there were no vehicles on the line and therefore the disopatcher removed the blocking of the signals. They did still not switch to green and the dispatcher then gave a oral permission to passenger train 7081 to pass the the red signal. When the passenger train arrived Brännland, the driver saw that the switch was unlocked and positioned towards the side track. He started braking the train and stopped in the switch. At the same time, the supervisor of movement 76910 was ‚pvomg tpwards tje switch. However, the supervisor detected the approaching passenger train and managed to stop the movement.”

Based on the scenario SE-511 it can be summarized, that a lack of up-to-date information and a lack of a separate control instance led to the accident. The dispatcher had no accurate information about the current state of the track. Manuel operations without any review procedure led to the accident.

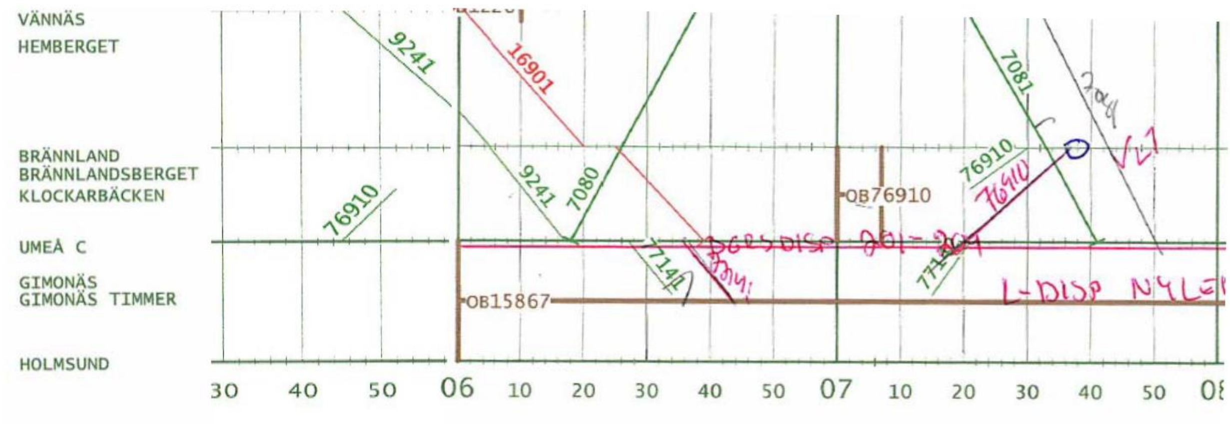


**Figure 9: Reconstruction of scenario SE-511**

It is interesting that the information situation is not completely clear, especially in sections where different tracks are brought together.

## 4.1.2. Accident report – PL-5985

| *Report Type* | *Investigation Status* | *ERAIL Occurrence* | *Title* | *Reporting Body* | *Date of occurrence* |
|---|---|---|---|---|---|
| **Final report** | **Closed** | **PL-5985** | **Trains collision, 19-05-19, Rybnik Towarowy (Poland)** | **State Commission on Rail Accident Investigation** | **2019-05-19** |
| "On 19.05.2019 at ca. 2:30 p.m., PKP CARGO S.A. railway operator reported to the RTB station master the need to have the locomotive located in the RTA shunting loop dispatched as helper locomotive for TMS 444255 freight train on route from Rybnik Towarowy to Chalupki – due to the profile of the section (climbing a gradient) and forecast of heavy rainfall – waiting on track 308 in the RTB and RTD run-around loop. At 2:58 p.m., at Rybnik Towarowy station, shunting locomotive SM42-1205 (6Dg) was dispatched from RTA shunting area into RTB positioning loop as helper locomotive for freight train TMS 444255 located on track 308. Without discussing the shunting manoeuvre with the driver of locomotive no. SM42-1205, the signaller at RTB signal box allowed its movement by the shunt signal indicating the signal 'shunting prohibited' by giving the following instruction: 'at 309 as helper for train to Chalupki... you can skip 257'. In doing so, the signaller wrongly directed the locomotive no. SM42-1205 to track 309, instead of 308. The locomotive arrived at a set of empty wagons located on this track at 3:07:30 p.m. (according to the time recorded in the monitoring system on locomotive ST48-049 located on track 308 at the front of freight train TMS 444255). The driver of locomotive SM42–1205, having reached the set of empty wagons on track 309, connected the locomotive with a screw and air coupling to the wagons located on that track. He then contacted the driver of locomotive ST48-049 located on track 308 at the front of freight train TMS 444255 by telephone (private mobile phone) to discuss the execution of a simplified brake test, which was not performed at all. Then, after the coupling, he made contact again by mobile phone with the driver of locomotive ST48-049 and informed him that he was connected to the train set and that they could proceed with a simplified brake test. After a failed attempt to perform the simplified test of the combined brake, the driver of the hauler locomotive instructed the driver of the pusher locomotive to disconnect the brake couplings (close the air valves) and remain connected to the set by means of the screw coupling only. The driver of locomotive ST48-049 at the front the train, having received the permission signal (S10) from the RTD command box transmitted on the T3082 exit signal from track 308, at 3:23:55 p.m. started the freight train TMS 444255 and instructed the driver of pusher locomotive SM42-1205, mistakenly located on track 309 at the end of the empty wagon train set, to go 'full ahead''. The driver of locomotive ST48-049 at the front of the TMS 444255 train was not authorised to give the instruction to the driver of the pusher locomotive; he was only required to inform him of starting the train. During the departure of the TMS 444255 freight train from track 308 at 3:25:10 p.m., a side collision took place at the fouling point of railroad switch 452 with the simultaneously pushed set of empty freight wagons from track 309. This resulted in | | | | | |

| |
|---|
| derailment of three empty wagons of the pushed train set from track 309 and damage to 8 freight wagons of the departing freight train TMS 444255 on route from Rybnik Towarowy to Chalupki. In addition, elements of the infrastructure were damaged (mechanical point machines for points 452 and 455 together with transmission routes, railroad switches 452 and 455 and eight wooden sleepers between switches 452 and 455 were destroyed). The speed of the pushed set of empty coal wagons at the time of the event was about 10 km/h, while the speed of the TMS 444255 train at the time of the event was 16 km/h.“ |

The dispatcher planned a mission and guided the train to the wrong track. In this scenario PL-5985 a supervision system for the route setting could work as well, to guide and support the dispatcher during the operation. A control instance had informed the dispatcher about the wrong execution. Here a faulty merging led to an accident as well.

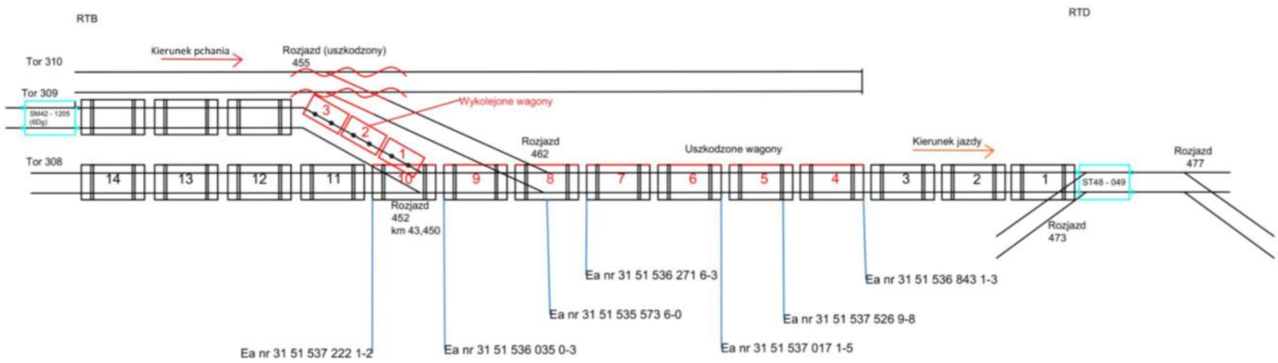


**Figure 10: Reconstruction of scenario PL-5985**

### 4.1.3. Accident report – NO-10230

| ***Report Type*** | ***Investigation Status*** | ***ERAIL Occurrence*** | ***Title*** | ***Reporting Body*** | ***Date of occurrence*** |
|---|---|---|---|---|---|
| **Notification** | **Open** | **NO-10230** | **Trains collision near miss, 11/05/2022, Bolna station** | **NIB NO** | **2022-05-19** |
| "On Wednesday 11.05.2022 a northbound passenger train had to initiate emergency brake to avoid collision with the last wagon in a crossing southbound freight train. Bolna station is a manned station with single signalling system. The freight train was taken into track number 2, before the train dispatcher operated the track switch into normal position (track 1). The passenger train ran through the station, and had to initiate emergency brake to avoid collision. The last wagon of the freight train stood in track 1, but inside the switch." | | | | | |

The potential accident was only identified by the train driver and was also preceded by an incorrect setting of the points, which could have been avoided by assistance during the operation. For this reason, the focus of possible example scenarios should be on passing tracks and rail junctions.

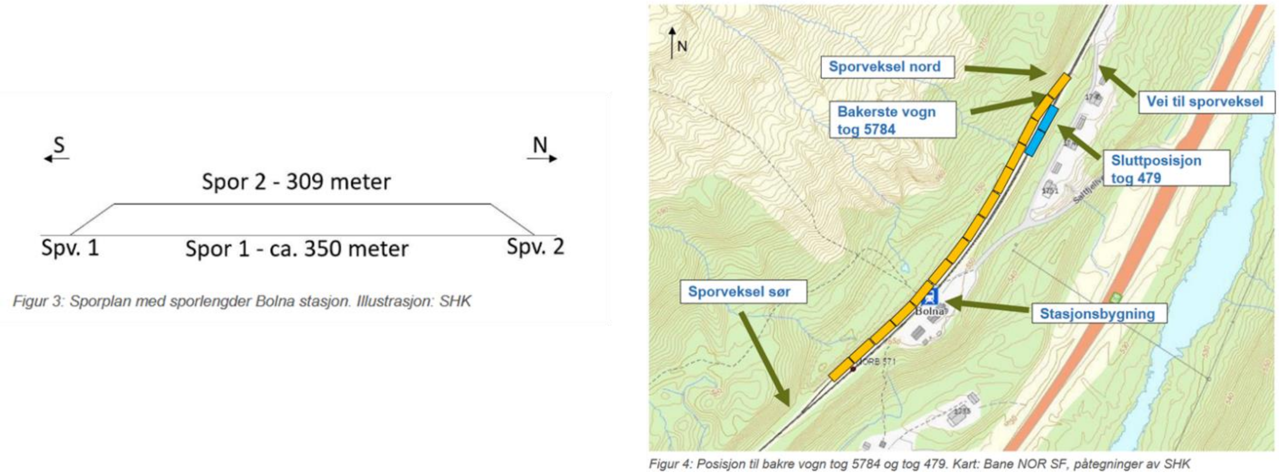


**Figure 11: Reconstruction of scenario NO-10230**

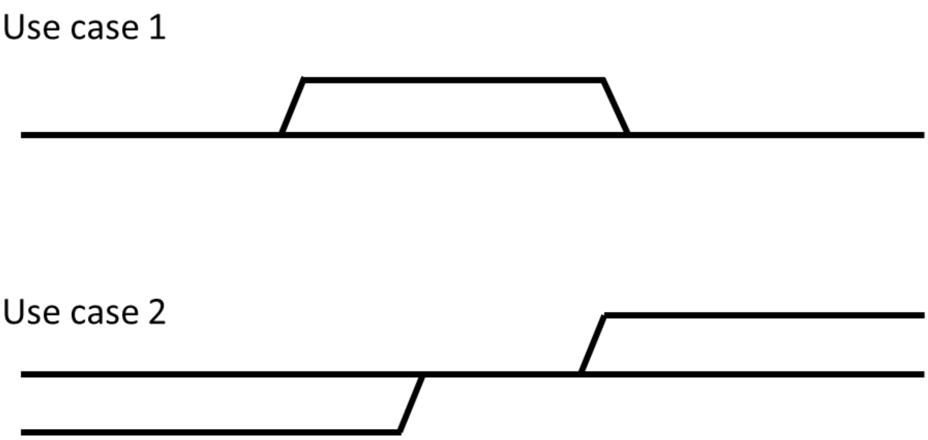


**Figure 12: Track network of derived use cases for the concept evaluation**

Figure 12 shows two simplified track layouts derived from the accident reports which include basic use cases that could lead to problems (accidents) or deadlocks (incidents). These challenging scenarios demonstrate that our concept is capable of handling real-world challenges, not just ideal conditions. If our system can solve deadlock problems we want to explain the innovative aspects of our work, making it clear what sets the approach apart from current solutions. Additionally, by explaining how our system works in difficult scenarios we hope to explain the main concept, and demonstrate the general idea. By applying our approach to these use cases we want to evaluate the practical value and how it can be useful in real-world applications, not just in theory. Including these use cases the result should be proving that our system is robust, innovative, trustworthy, and practically valuable.

The first use case is a passing loop, also known as a passing siding or crossing loop, is a section of track on a single-line railway that allows trains traveling in opposite directions to pass each other. Since the main line only has one track, trains cannot pass each other directly without causing delays. The passing loop solves this problem by providing a short, parallel track where one train can wait while the other passes.

Imagine two trains approaching each other on the same track. One train will enter the passing loop and wait on the side track while the other continues on the main track. After the second train passes, the waiting train can rejoin the main track and continue its journey.

The main challenge in coordinating the use of passing loops is timing. Both trains need to arrive at the loop at the right time to avoid delays. If the timing is off, one train might have to wait longer than necessary, which can disrupt the schedule and cause a ripple effect of delays across the network.

Communication between train operators and control centers is crucial to ensure smooth operation. They need to be aware of each train's location, speed, and schedule to make real-time decisions about which train should use the loop and when. The complexity increases with more trains or if there are unexpected delays, making precise coordination essential to keeping the trains running efficiently. So, the key challenges involve careful timing, effective communication, and quick decision-making to avoid any bottlenecks or delays that could affect the entire railway network.

The second use case is a simple rail junction, where several scenarios can be analysed to demonstrate how the AnRS manages trains to switch from one line to another, enabling more complex routes and connections. The primary challenge in managing train junctions is ensuring the safe and efficient movement of trains through these intersections without causing delays or accidents. Managing a train junction is all about coordinating the movement of trains in a way that balances efficiency with safety. This requires meticulous planning, real-time communication, and reliable technology to keep trains running smoothly and on time through these critical parts of the railway network. The primary challenge in managing train junctions is ensuring the safe and efficient movement of trains through these intersections without causing delays or accidents. This involves several key aspects:

## 4.2. Requirements

The following breakdown was made for the requirements survey: Necessary Environmental conditions/information, existing Communication Networks, Interlocking and Signalling Systems, Safety and Emergency regulations, Control Logic and Decision-Making rules. These fields could be linked to the following Components of the Traffic Management System: Interlocking, Information Management, Capturing System, AnRS system. The Figure 13 shows the relevant and considered requirements[11] for the AnRS system.

[11] Derived from Chapter 3.3

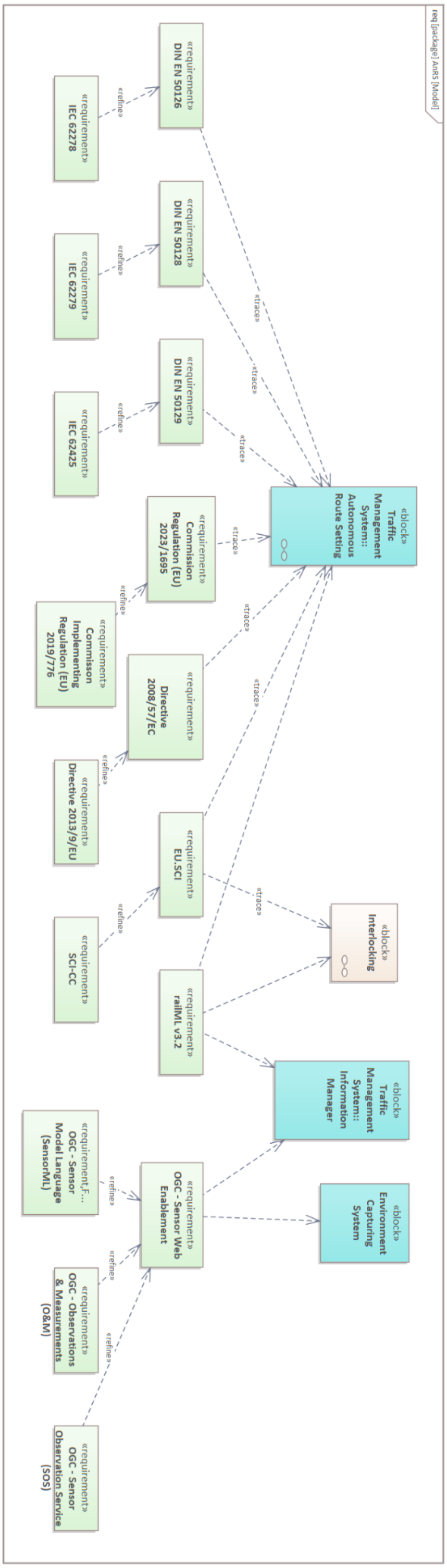


**Figure 13: Requirements diagram for AnRS**

Main focus of requirements is on interfaces and interoperability, to guarantee a system that can be integrated into the existing rail infrastructure.

### 4.2.1. railML and Interlocking

railML, or Railway Markup Language, is a standardized data exchange format based on XML, designed specifically for the railway industry. One of its critical components is the interlocking schema, which focuses on the systems that control railway signals and switches to ensure safe train movements (cf. Figure 5).

The railML interlocking schema standardizes the representation of interlocking data, enabling interoperability and efficient data exchange between different railway systems and organizations. This standardization ensures that various interlocking systems can communicate effectively, leading to enhanced safety and operational efficiency.

The interlocking schema includes several key components. Signals are described in detail, including their types, positions, and aspects. Information about switches, such as their locations, states, and control mechanisms, is standardized. Definitions of routes are included, covering the sequence of track sections and switches that form a safe path for a train. Detailed data about individual track sections, including their boundaries and connectivity, is provided. Interlocking logic, which encompasses the rules and conditions that govern the operation of signals and switches to ensure safe train movements, is also a crucial part of the schema.

The data exchange process involves the creation of interlocking data in the railML format using compatible software tools. This data includes the configuration and operational rules of the interlocking system. The railML files containing interlocking data are then shared between different systems, such as traffic management systems, signalling control centers, and maintenance applications. The receiving systems import the railML files and integrate the interlocking data into their own databases, ensuring consistent and accurate information across platforms.

### 4.2.2. SCI-CC from Eulynx

The System Context Interface - Command and Control (SCI-CC) from EULYNX is a specification aimed at standardizing the interface between central command systems and field elements like signals and switches within European railway networks. It promotes interoperability by enabling different signalling and control systems to communicate effectively regardless of the manufacturer.

### 4.2.3. IEC

IEC 62278, IEC 62279, and IEC 62425 are significant standards that collectively provide a framework for the safety and reliability of railway signalling and control systems, including interlocking. IEC 62278, also known as EN 50126, outlines the requirements for Reliability, Availability, Maintainability, and Safety (RAMS) in railway applications. It provides a comprehensive lifecycle approach for managing these aspects in railway systems, ensuring they meet stringent safety and performance criteria from conception through to decommissioning. For interlocking systems, this means that RAMS principles must be applied to ensure that the system is reliable, available when needed, maintainable, and safe to operate throughout its lifecycle.

IEC 62279, also known as EN 50128, specifies the requirements for the development, deployment, and maintenance of software for railway control and protection systems. This standard is critical for interlocking systems, which rely heavily on software to manage the safe movement of trains. It covers the software development process, including risk assessment, quality assurance, validation, and verification, ensuring that the software used in interlocking systems is robust, reliable, and safe.

IEC 62425, also known as EN 50129, defines the safety-related requirements for railway signalling systems, including interlocking. It provides guidelines for the safety assessment and certification of signalling systems to ensure they meet the necessary safety integrity levels (SIL). For interlocking systems, this standard is crucial as it ensures that the signalling logic, hardware, and software components meet the required safety standards to prevent accidents and ensure safe train operations.

Interlocking systems, which control railway signals and switches to prevent conflicting train movements, must comply with these standards to ensure safety and reliability. The interlocking system's lifecycle is guided by IEC 62278 (RAMS), which ensures that the system is designed, maintained, and operated with reliability, availability, maintainability, and safety in mind. The software that controls the interlocking system must adhere to IEC 62279, ensuring that it is developed and maintained to high standards of safety and reliability. Finally, the overall safety requirements and safety integrity levels of the interlocking system must comply with IEC 62425, ensuring that all components of the interlocking system meet the necessary safety standards.

### 4.2.4. Commission Regulation

Commission Regulation 2023/1695 and 2019/776 are important regulations that relate to the standardization and interoperability of railway systems within the European Union, including signalling and control systems such as interlocking. These regulations aim to ensure that railway systems across different countries can work together seamlessly, enhancing safety, efficiency, and reliability.

Commission Regulation 2016/919, also known as the Technical Specification for Interoperability relating to the Control-Command and Signalling (CCS TSI), sets out the requirements for the European Rail Traffic Management System (ERTMS). ERTMS is a standardized system for managing and controlling train movements across Europe, designed to replace the various national signalling systems with a single, harmonized system. This regulation specifies the functional and technical requirements for ERTMS components, including the European Train Control System (ETCS) and the Future Railway Mobile Communication System (FRMCS; former Global System for Mobile Communications-Railway (GSM-R)). For interlocking systems, this regulation ensures that they are compatible with ERTMS standards, enabling trains to operate safely and efficiently across borders without changing signalling systems.

Commission Regulation 2019/776, also known as the Common Safety Methods for Risk Evaluation and Assessment (CSM RA), establishes the framework for assessing and managing risks associated with changes in the railway system. It provides guidelines for id identifying hazards, assessing risks, and implementing measures to mitigate those risks to an acceptable level. This regulation applies to any significant change in the railway system, including modifications to interlocking systems. It ensures that changes to interlocking systems are thoroughly evaluated for safety risks and that appropriate mitigation measures are implemented.

### 4.2.5. Directives

Directive 2016/797/EC and Directive 2016/798/EU are key legislative acts that relate to the interoperability and safety of railway systems within the European Union, including signalling and control systems such as interlocking. These directives aim to harmonize technical standards and facilitate the seamless operation of trains across different national railway networks in Europe.

Directive 2008/57/EC, known as the Interoperability of the Rail System within the Community, establishes the conditions that must be met to achieve interoperability within the European rail system. This directive covers all aspects of railway infrastructure and rolling stock, ensuring that they are compatible with each other across different EU member states. For interlocking systems, this directive requires that they meet specific interoperability specifications to ensure safe and efficient cross-border railway operations. The directive sets out essential requirements for safety, reliability, and technical compatibility, which interlocking systems must adhere to. This harmonization helps to remove technical barriers to international rail transport, facilitating smoother and more efficient cross-border rail services.

Directive 2013/9/EU, which amends Directive 2008/57/EC, focuses on further enhancing the interoperability of the European rail system. It updates certain technical specifications and requirements to reflect technological advancements and changes in the railway industry. This directive includes specific provisions for the ERTMS, ensuring that signalling and control systems, including interlocking, comply with the latest ERTMS standards. By mandating adherence to these updated specifications, Directive 2013/9/EU ensures that interlocking systems continue to support safe and efficient train operations across Europe.

## 4.3. Function Specification - Operational Analysis[12]

For the derivation of the AnRS concept and system architecture the ARCADIA method and Capella, as proposed by the Europe's Rail, was used. ARCADIA (Architecture Analysis & Design Integrated Approach) is a model-based systems engineering (MBSE) method designed by Thales. It helps engineers and architects develop complex systems by emphasizing stakeholder needs, system functionalities, and operational requirements. It promotes a collaborative approach through visual models to ensure system consistency and traceability throughout its lifecycle.

The Capella tool is an open-source, model-based engineering tool that implements the ARCADIA method. It facilitates systems modelling by providing graphical representations of the system's architecture at various levels, from operational to physical.

The following Operational Analysis focuses on understanding the stakeholders' needs and the operational context. This phase involves defining operational scenarios and ensuring the system meets high-level mission objectives without specifying technical solutions.

### 4.3.1. Description of the Operational entities

The following figure shows the distribution of the entities and actors in the Operational Context of the System Design:

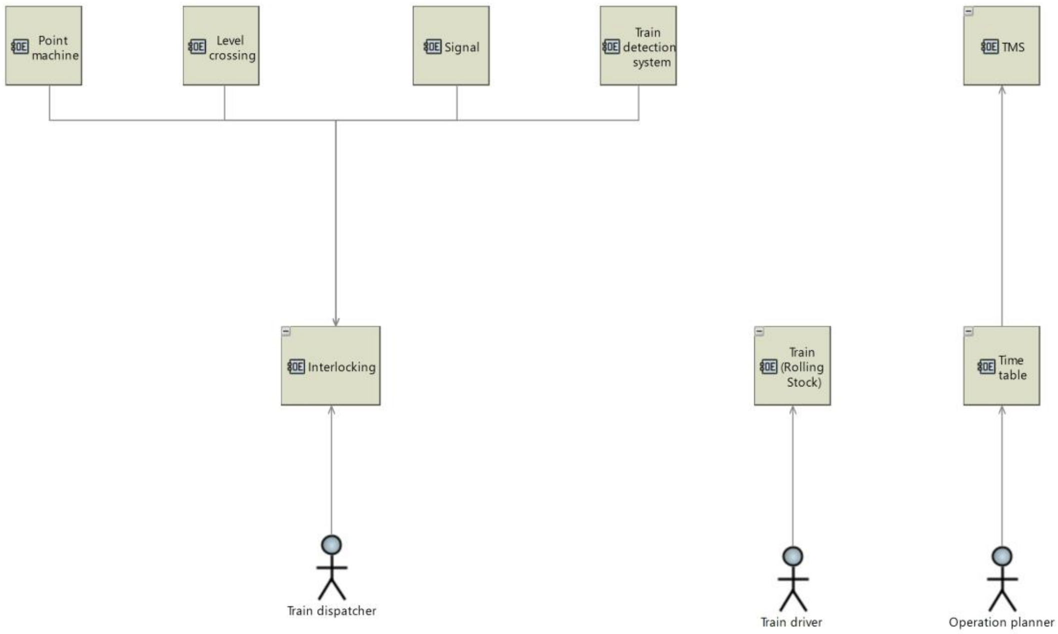


**Figure 14: Operational Entities Diagram**

In the following subchapters, the description of the actors will be displayed.

[12] generated automatically by M2Doc from Capella model of the system

#### 4.3.1.1. Entity: Train (Rolling Stock)

Description:

*No description*

#### 4.3.1.2. Entity: Train driver

Description:

*No description*

#### 4.3.1.3. Entity: Infrastructure

Description:

Sensor and communication network to observe the trackside

#### 4.3.1.4. Entity: Interlocking

Description:

*No description*

#### 4.3.1.5. Entity: Train dispatcher

Description:

*No description*

#### 4.3.1.6. Entity: Signal

Description:

*No description*

#### 4.3.1.7. Entity: Point machine

Description:

*No description*

#### 4.3.1.8. Entity: Train detection system

Description:

*No description*

#### 4.3.1.9. Entity: Level crossing

Description:

*No description*

#### 4.3.1.10. Entity: TMS

Description:

*No description*

### 4.3.1.11. Entity: Time table

Description:

*No description*

### 4.3.1.12. Entity: Operation planner

Description:

*No description*

## 4.3.2. Description of the Operational capabilities

Operational analysis focuses on the interaction and cooperation between the train driver, the train dispatcher and operation planner. The following chapters show all the entities operationally related to the AnRS system.

The next step in the operational modelling of the system is to identify the operational capabilities of the system and its relationship with the previously identified actors. The relevant existing capabilities should not be affected by the new system. These are primarily the optimization of the traffic flow while maintaining safety and the possible capacities as boundary conditions. Within this area of tension, the AnRS should enable Conflict Resolution and Dynamic Route Changing.

The following capabilities are then identified:

- Enhances Safety: enhances safety by automating route-setting processes, minimizing the risk of human errors. It ensures that trains follow designated paths, adhere to speed limits, and avoid potential hazards, reducing the likelihood of accidents.
- Capacity Management: efficient capacity management by allocating routes that maximize the use of available track resources. This is particularly important in busy rail networks to handle increasing traffic demands without compromising safety.
- Optimize Train Flow: optimizing the movement of trains by determining the most efficient and conflict-free routes. This minimizes delays, reduces congestion, and enhances overall system efficiency.
- Dynamic Route Changes: dynamically adapt to changes in operational conditions, such as track maintenance, emergencies, or unexpected events. This flexibility allows for real-time adjustments to routes, minimizing disruptions and optimizing the use of available infrastructure.
- Conflict Resolution: help in preventing conflicts between trains by intelligently selecting routes that avoid potential clashes. It considers factors like train speed, braking distances, and track occupancy to ensure safe and conflict-free operations.

To identify stakeholder needs, two capabilities dynamic route changing, and conflict resolution were examined in more detail and the existing activities were defined.

The first step in conflict resolution is to recognize potential conflicts on both the supply and demand sides. A further process here is the resolution of conflicts and the communication to resolve these conflicts.

To finish the operational modeling, it is necessary to define the following scenarios for the capabilities analyzed. In them you can see the temporal order of activities in relation to the entities involved.

#### 4.3.2.1. Operational Capabilities Description

##### *4.3.2.1.1. Capability: Optimize Train Flow*

Description:

Capability is about optimizing the movement of trains by determining the most efficient and conflict-free routes. This minimizes delays, reduces congestion, and enhances overall system efficiency.

*This capability is not involved in any scenario.*

##### *4.3.2.1.2. Capability: Dynamic Route Changes*

Description:

Capability to dynamically adapt to changes in operational conditions, such as track maintenance, emergencies, or unexpected events. This flexibility allows for real-time adjustments to routes, minimizing disruptions and optimizing the use of available infrastructure.

This capability is involved in the following scenarios:

- [OES] Dynamic Route Changes

##### *4.3.2.1.3. Capability: Conflict Resolution*

Description:

Capability about help in preventing conflicts between trains by intelligently selecting routes that avoid potential clashes. It considers factors like train speed, braking distances, and track occupancy to ensure safe and conflict-free operations.

This capability is involved in the following scenarios:

- [OES] Conflict Resolution

##### *4.3.2.1.4. Capability: Capacity Management*

Description:

Capability about efficient capacity management by allocating routes that maximize the use of available track resources. This is particularly important in busy rail networks to handle increasing traffic demands without compromising safety.

*This capability is not involved in any scenario.*

#### *4.3.2.1.5. Capability: Predictive Analytics*

Description:

predictive analytics to anticipate potential issues or delays. By analyzing historical data and current conditions, the system can proactively suggest optimal routes and strategies to enhance overall efficiency.

*This capability is not involved in any scenario.*

#### *4.3.2.1.6. Capability: Enhance Safety*

Description:

enhances safety by automating route-setting processes, minimizing the risk of human errors. It ensures that trains follow designated paths, adhere to speed limits, and avoid potential hazards, reducing the likelihood of accidents.

*This capability is not involved in any scenario.*

### 4.3.2.2. Capabilities/Entities Relationships

The following figure shows the relations between capabilities and entities:

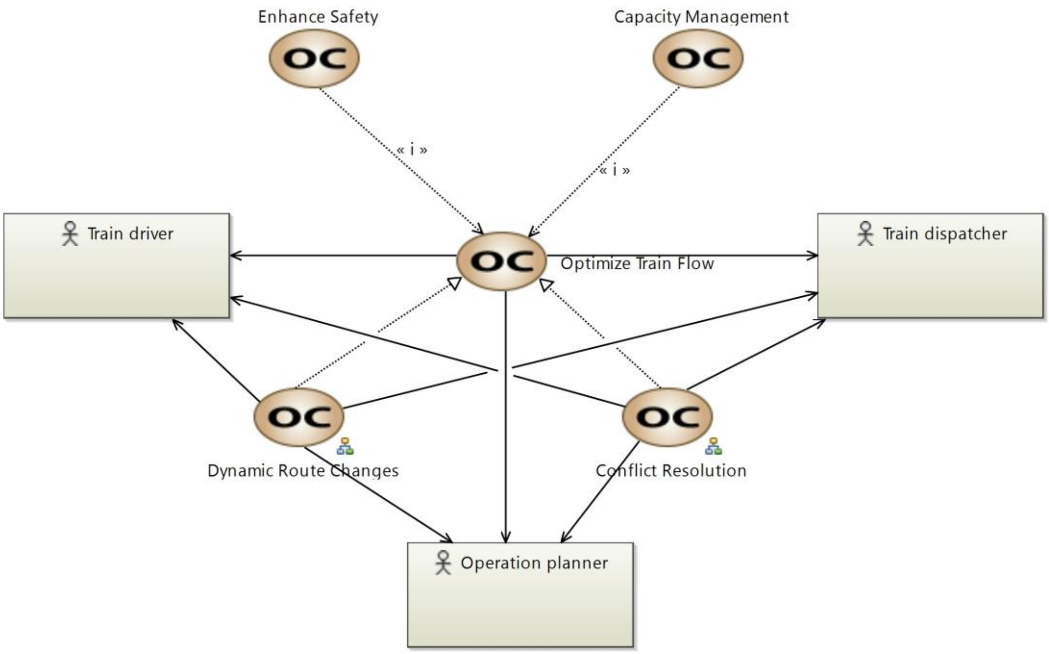


**Figure 15: Operational Capabilities Diagram**

### 4.3.3. Description of the Activities

The activities are sorted in the following functions:

- Preparing the train
- Identifying with information
- Assign a route
- Deciding for a route
- Reviewing the route request
- Authorizing the selected route
- Preparing the journey
- Track the trains progress
- Start the mission
- Control speed
- Adherence to signal indications
- Analysis for improvement of future operations
- Post-processing of the trip
- Observe network
- Identifying conflict (track)
- Identifying conflict (plan)
- Identifying conflict (train)
- Managing/resolving conflict
- Analyzing (severity of the) conflict
- Adjust the routes of one or more trains
- Processing conflict resolution
- Following new route (instructions)
- Observing new movements
- Carrying out conflict avoidance
- Adhere to new instructions
- Update the plan to avoid future conflicts
- Update the plan
- Select the (planned) route

The diagrams of the activities for each function are shown below:

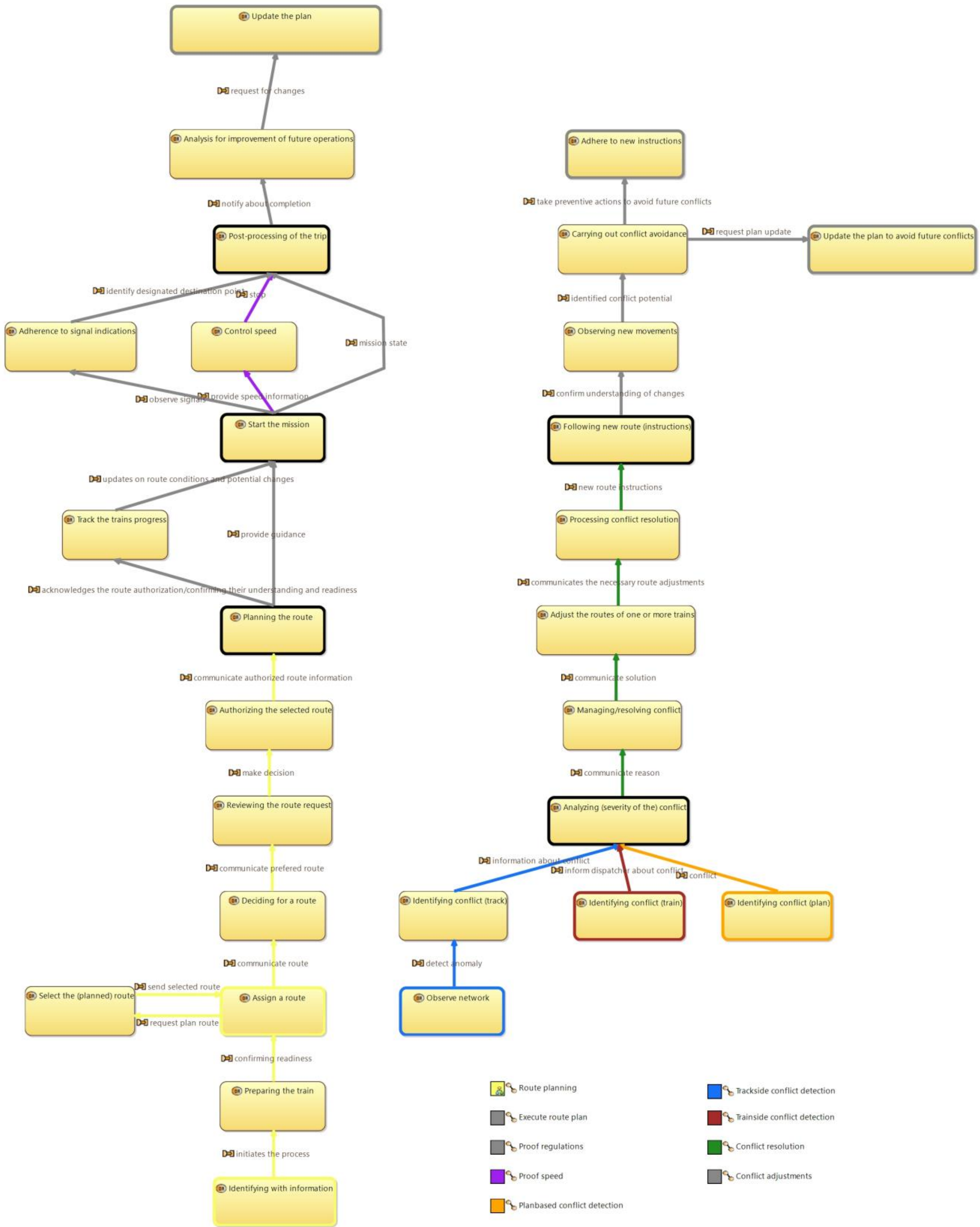


**Figure 16: Operational Activity Interaction Diagram**

#### 4.3.3.1. Activity: Preparing the train

Description:

technical preparation of the train

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| initiates the process | Identifying with information |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| confirming readiness | Assign a route |

#### 4.3.3.2. Activity: Identifying with information

Description:

Each train is identified in the system, typically through a unique identifier or code. The train driver initiates the process by logging into the system, providing necessary identification details, and confirming the train readiness.

List of inputs:

*The activity has no input associated*

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| initiates the process | Preparing the train |

### 4.3.3.3. Activity: Assign a route

Description:

Based on the schedule and network conditions, the dispatcher or central control system may automatically assign a route to the train.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| confirming readiness | Preparing the train |
| send selected route | Select the (planned) route |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| communicate route | Deciding for a route |
| request plan route | Select the (planned) route |

### 4.3.3.4. Activity: Deciding for a route

Description:

The train driver may request a specific route if there are multiple options available.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| communicate route | Assign a route |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| communicate preferred route | Reviewing the route request |

### 4.3.3.5. Activity: Reviewing the route request

Description:

The dispatcher reviews the route request, considering factors such as track availability, maintenance schedules, and other operational constraints. Once approved, the dispatcher authorizes the selected route for the train.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| communicate preferred route | Deciding for a route |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| make decision | Authorizing the selected route |

### 4.3.3.6. Activity: Authorizing the selected route

Description:

The dispatcher communicates the authorized route information to the train driver. The train operator acknowledges the route authorization, confirming their understanding and readiness to proceed along the assigned route.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| make decision | Reviewing the route request |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| communicate authorized route information | Preparing the journey |

### 4.3.3.7. Activity: Preparing the journey

Description:

The trains on-board control system receives the authorized route information and may provide guidance to the train operator. Automated train control systems may take over specific functions, such as speed control and adherence to signal indications.

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| communicate authorized route information | Authorizing the selected route |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
| --- | --- |
| provide guidance | Start the mission |
| acknowledges the route authorization/confirming their understanding and readiness | Track the trains progress |

#### *4.3.3.8. Activity: Track the trains progress*

Description:

Both the dispatcher and the train operator have access to real-time monitoring systems. The dispatcher can track the trains progress, and the train operator can receive updates on route conditions and potential changes.

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| acknowledges the route authorization/confirming their understanding and readiness | Preparing the journey |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| updates on route conditions and potential changes | Start the mission |

### 4.3.3.9. Activity: Start the mission

Description:

The train driver starts the journey

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| provide guidance | Preparing the journey |
| updates on route conditions and potential changes | Track the trains progress |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| provide speed information | Control speed |
| observe signals | Adherence to signal indications |
| mission state | Post-processing of the trip |

### 4.3.3.10. Activity: Control speed

Description:

*No description*

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| provide speed information | Start the mission |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| stop | Post-processing of the trip |

### 4.3.3.11. Activity: Adherence to signal indications

Description:

Signals along the track indicate to the train operator whether the route is clear, occupied, or if there are any speed restrictions.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| observe signals | Start the mission |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| identify designated destination point | Post-processing of the trip |

### 4.3.3.12. Activity: Analysis for improvement of future operations

Description:

After the conflict is resolved, the dispatcher and operational staff may conduct a post-resolution analysis. This involves reviewing the incident, understanding the root causes, and identifying any systemic improvements that can be made to enhance future conflict management.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| notify about completion | Post-processing of the trip |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| request for changes | Update the plan |

### 4.3.3.13. Activity: Post-processing of the trip

Description:

documentation

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| identify designated destination point | Adherence to signal indications |
| stop | Control speed |
| mission state | Start the mission |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| notify about completion | Analysis for improvement of future operations |

### 4.3.3.14. Activity: Observe network

Description:

various sensors, train positioning data, and real-time monitoring

List of inputs:

*The activity has no input associated*

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| detect anomaly | Identifying conflict (track) |

### 4.3.3.15. Activity: Identifying conflict (track)

Description:

identifying conflicts based on the information available on the trackside.

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| detect anomaly | Observe network |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
| --- | --- |
| information about conflict | Analyzing (severity of the) conflict |

#### *4.3.3.16. Activity: Identifying conflict (plan)*

Description:

identifying conflicts based on the current and planned positions of trains

List of inputs:

*The activity has no input associated*

List of outputs:

| Outgoing Interaction | Outgoing Activity |
| --- | --- |
| conflict | Analyzing (severity of the) conflict |

#### *4.3.3.17. Activity: Identifying conflict (train)*

Description:

identifying conflicts based on the local information inside the train.

List of inputs:

*The activity has no input associated*

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| inform dispatcher about conflict | Analyzing (severity of the) conflict |

### 4.3.3.18. Activity: Managing/resolving conflict

Description:

could be: rerouting a train, delaying its departure, or holding it at a specific location until the conflict is resolved

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| communicate reason | Analyzing (severity of the) conflict |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| communicate solution | Adjust the routes of one or more trains |

### 4.3.3.19. Activity: Analysing (severity of the) conflict

Description:

considering the speeds of the involved trains, the distance between them, and the type of track configuration contributing to the conflict.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| inform dispatcher about conflict | Identifying conflict (train) |
| conflict | Identifying conflict (plan) |
| information about conflict | Identifying conflict (track) |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| communicate reason | Managing/resolving conflict |

### 4.3.3.20. Activity: Adjust the routes of one or more trains

Description:

The dispatcher may manually or automatically adjust the routes of one or more trains involved in the conflict. This could involve rerouting a train, delaying its departure, or holding it at a specific location until the conflict is resolved.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| communicate solution | Managing/resolving conflict |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| communicates the necessary route adjustments | Processing conflict resolution |

### 4.3.3.21. Activity: Processing conflict resolution

Description:

The dispatcher communicates the necessary route adjustments to the affected train operators. This communication is crucial to ensuring that train operators are aware of the changes and can adhere to the new route instructions.

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| communicates the necessary route adjustments | Adjust the routes of one or more trains |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
| --- | --- |
| new route instructions | Following new route (instructions) |

#### 4.3.3.22. Activity: Following new route (instructions)

Description:

The train driver follows the new route

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| new route instructions | Processing conflict resolution |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
| --- | --- |
| confirm understanding of changes | Observing new movements |

#### 4.3.3.23. Activity: Observing new movements

Description:

Throughout the conflict resolution process, the central control system continues to monitor the updated positions and movements of the trains involved. This ongoing monitoring helps ensure that the conflict is effectively resolved.

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| confirm understanding of changes | Following new route (instructions) |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
| --- | --- |
| identified conflict potential | Carrying out conflict avoidance |

### 4.3.3.24. Activity: Carrying out conflict avoidance

Description:

*No description*

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| identified conflict potential | Observing new movements |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
|---|---|
| take preventive actions to avoid future conflicts | Adhere to new instructions |
| request plan update | Update the plan to avoid future conflicts |
| Interaction 31 | FIP 2 |
| Interaction 32 | FIP 3 |

### 4.3.3.25. Activity: Adhere to new instructions

Description:

*No description*

List of inputs:

| Incoming Interaction | Incoming Activity |
|---|---|
| take preventive actions to avoid future conflicts | Carrying out conflict avoidance |

List of outputs:

*The activity has no output associated*

### 4.3.3.26. Activity: Update the plan to avoid future conflicts

Description:

*No description*

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| request plan update | Carrying out conflict avoidance |

List of outputs:

*The activity has no output associated*

#### 4.3.3.27. Activity: Update the plan

Description:

*No description*

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| request for changes | Analysis for improvement of future operations |

List of outputs:

*The activity has no output associated*

#### 4.3.3.28. Activity: Select the (planned) route

Description:

*No description*

List of inputs:

| Incoming Interaction | Incoming Activity |
| --- | --- |
| request plan route | Assign a route |

List of outputs:

| Outgoing Interaction | Outgoing Activity |
| --- | --- |
| send selected route | Assign a route |

### 4.3.4. Description of Operational Processes

#### *4.3.4.1. Operational Process: Trackside conflict detection*

Description:

The trackside uses sensors and communication infrastructure with information processing to detect potential conflicts and anomalies

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| detect anomaly | Observe network | Identifying conflict (track) |
| information about conflict | Identifying conflict (track) | Analyzing (severity of the) conflict |

Involving capabilities[13]:

*No involving capability*

#### *4.3.4.2. Operational Process: Trainside conflict detection*

Description:

The onboard system uses sensors and communication infrastructure with information processing to detect potential conflicts and anomalies

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| inform dispatcher about conflict | Identifying conflict (train) | Analyzing (severity of the) conflict |

Involving capabilities:

[13] In Capella, operational processes can be modelled using various diagrams. However, the current analysis did not explore this in depth, as the deliverable primarily serves as a conceptual study. Consequently, no capabilities were involved at this stage.

*No involving capability*

#### 4.3.4.3. Operational Process: Conflict resolution

<u>Description:</u>

based on the available information the conflict will be resolved without violencing the safety and efficiency criterias

<u>Involved elements:</u>

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| communicate reason | Analyzing (severity of the) conflict | Managing/resolving conflict |
| communicate solution | Managing/resolving conflict | Adjust the routes of one or more trains |
| communicates the necessary route adjustments | Adjust the routes of one or more trains | Processing conflict resolution |
| new route instructions | Processing conflict resolution | Following new route (instructions) |

<u>Involving capabilities:</u>

*No involving capability*

#### 4.3.4.4. Operational Process: Route planning

This operational process is illustrated by the following diagram:

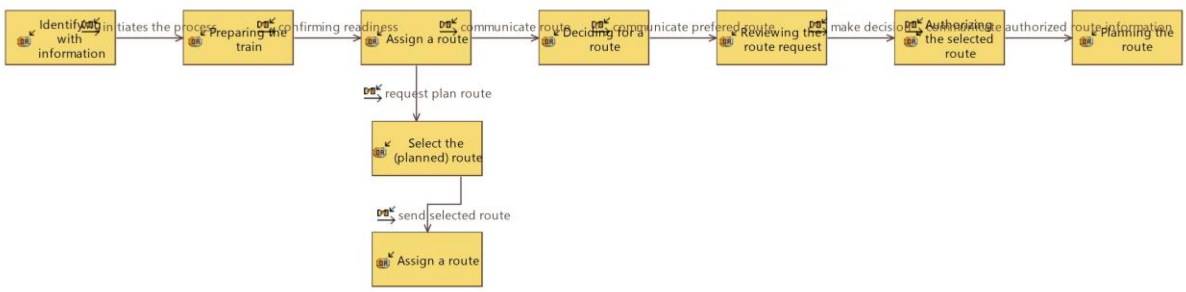


**Figure 17: Operational Process Diagram**

<u>Description:</u>

initial and strategic route planning for new trains befor journey

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| initiates the process | Identifying with information | Preparing the train |
| confirming readiness | Preparing the train | Assign a route |
| communicate route | Assign a route | Deciding for a route |
| communicate prefered route | Deciding for a route | Reviewing the route request |
| make decision | Reviewing the route request | Authorizing the selected route |
| communicate authorized route information | Authorizing the selected route | Preparing the journey |
| request plan route | Assign a route | Select the (planned) route |
| send selected route | Select the (planned) route | Assign a route |

Involving capabilities:

*No involving capability*

#### 4.3.4.5. Operational Process: Proof regulations

Description:

check the current/local regulations against behaviour

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| observe signals | Start the mission | Adherence to signal indications |
| identify designated destination point | Adherence to signal indications | Post-processing of the trip |

Involving capabilities:

*No involving capability*

#### 4.3.4.6. Operational Process: Proof speed

Description:

check the allowed speed against behaviour

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| provide speed information | Start the mission | Control speed |
| stop | Control speed | Post-processing of the trip |

Involving capabilities:

*No involving capability*

#### 4.3.4.7. Operational Process: Planbased conflict detection

Description:

The plan uses information processing to detect potential conflicts and anomalies

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| conflict | Identifying conflict (plan) | Analyzing (severity of the) conflict |

Involving capabilities:

*No involving capability*

#### 4.3.4.8. Operational Process: Conflict adjustments

Description:

analyze and communicate adjustments to guarantee safety

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| confirm understanding of changes | Following new route (instructions) | Observing new movements |
| identified conflict potential | Observing new movements | Carrying out conflict avoidance |
| request plan update | Carrying out conflict avoidance | Update the plan to avoid future conflicts |
| take preventive actions to avoid future conflicts | Carrying out conflict avoidance | Adhere to new instructions |

Involving capabilities:

*No involving capability*

#### 4.3.4.9. Operational Process: Execute route plan

Description:

movement from A to B

Involved elements:

This functional process involves the following elements:

| Functional exchange | Source activity | Target activity |
|---|---|---|
| provide guidance | Preparing the journey | Start the mission |
| acknowledges the route authorization/confirming their understanding and readiness | Preparing the journey | Track the trains progress |
| updates on route conditions and potential changes | Track the trains progress | Start the mission |
| mission state | Start the mission | Post-processing of the trip |

| Functional exchange | Source activity | Target activity |
| --- | --- | --- |
| notify about completion | Post-processing of the trip | Analysis for improvement of future operations |
| request for changes | Analysis for improvement of future operations | Update the plan |

Involving capabilities:

*No involving capability*

### 4.3.5. Description of Scenarios

#### *4.3.5.1. Scenario Functions: Optimize Train Flow*

Description:

Capability is about optimizing the movement of trains by determining the most efficient and conflict-free routes. This minimizes delays, reduces congestion, and enhances overall system efficiency.

*No diagrams defined in the scenario.*

#### *4.3.5.2. Scenario Functions: Dynamic Route Changes*

Description:

Capability to dynamically adapt to changes in operational conditions, such as track maintenance, emergencies, or unexpected events. This flexibility allows for real-time adjustments to routes, minimizing disruptions and optimizing the use of available infrastructure.

The following figures show the Activities interaction diagram (OAS) and the Entities Interaction Diagrams (OES) of the scenario.

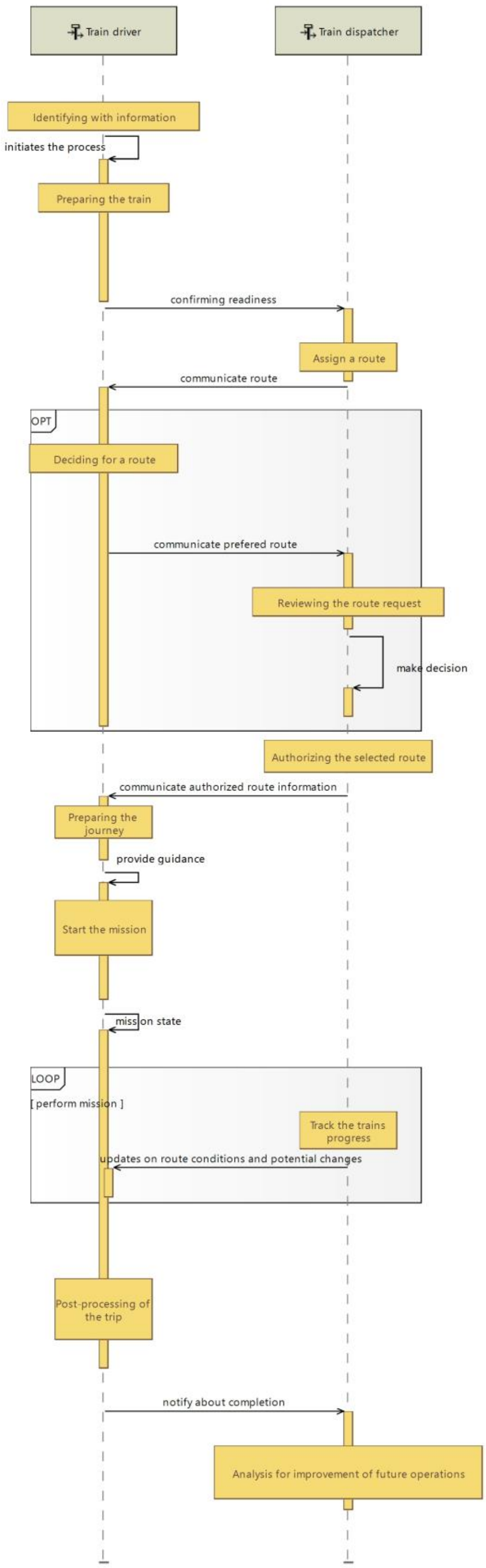


**Figure 18: Operational Activity Interaction Diagram**

#### 4.3.5.3. Scenario Functions: Conflict Resolution

Description:

Capability about help in preventing conflicts between trains by intelligently selecting routes that avoid potential clashes. It considers factors like train speed, braking distances, and track occupancy to ensure safe and conflict-free operations.

The following figures show the Activities interaction diagram (OAS) and the Entities Interaction Diagrams (OES) of the scenario.

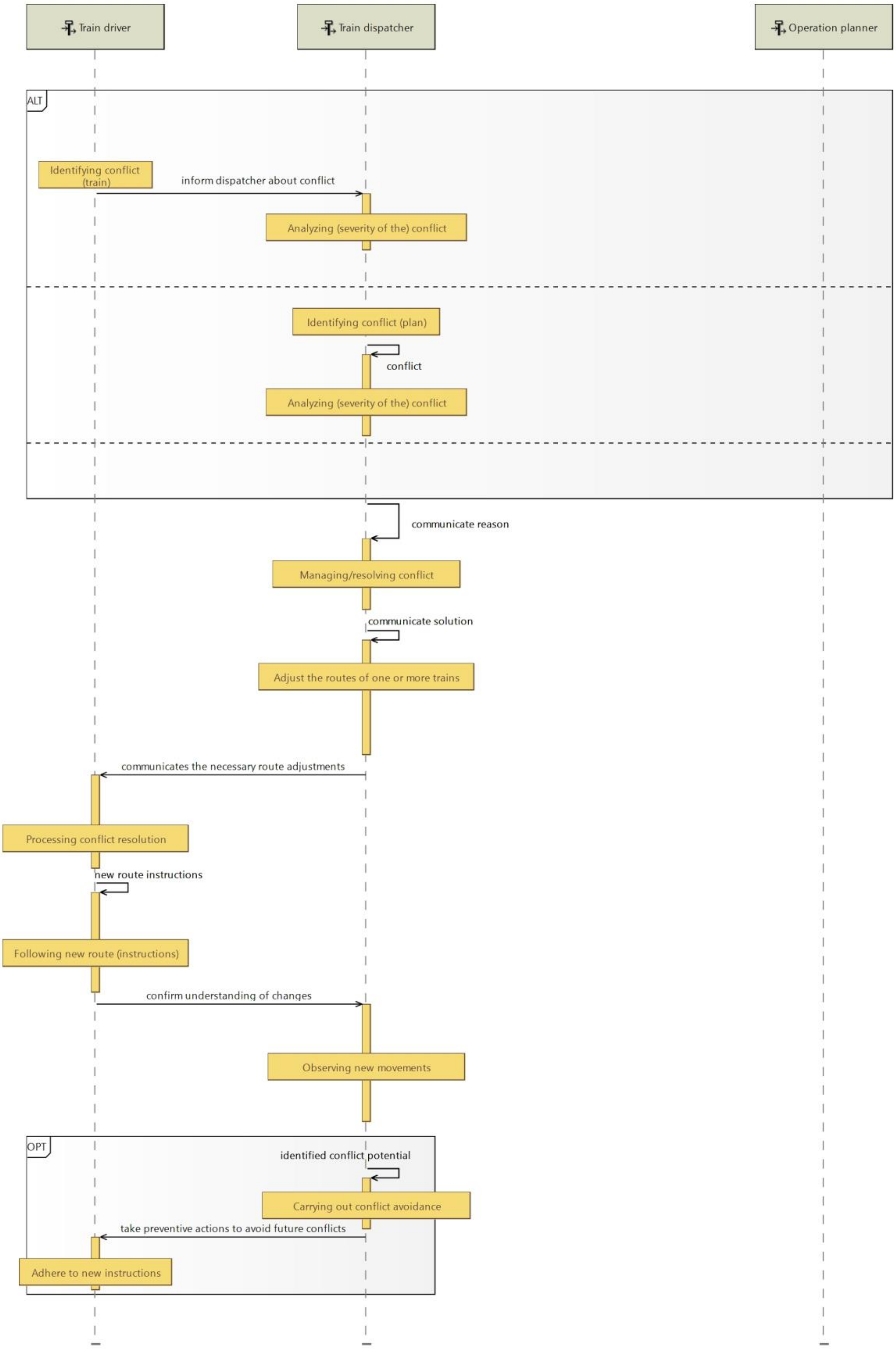


**Figure 19: Operational Activity Interaction Diagram**

#### 4.3.5.4. Scenario Functions: Capacity Management

Description:

Capability about efficient capacity management by allocating routes that maximize the use of available track resources. This is particularly important in busy rail networks to handle increasing traffic demands without compromising safety.

*No diagrams defined in the scenario.*

#### 4.3.5.5. Scenario Functions: Predictive Analytics

Description:

predictive analytics to anticipate potential issues or delays. By analyzing historical data and current conditions, the system can proactively suggest optimal routes and strategies to enhance overall efficiency.

*No diagrams defined in the scenario.*

#### 4.3.5.6. Scenario Functions: Enhance Safety

Description:

enhances safety by automating route-setting processes, minimizing the risk of human errors. It ensures that trains follow designated paths, adhere to speed limits, and avoid potential hazards, reducing the likelihood of accidents.

*No diagrams defined in the scenario.*

## 4.3.6. Description of the Operational Context and its Environment

### 4.3.6.1. Architecture Diagram

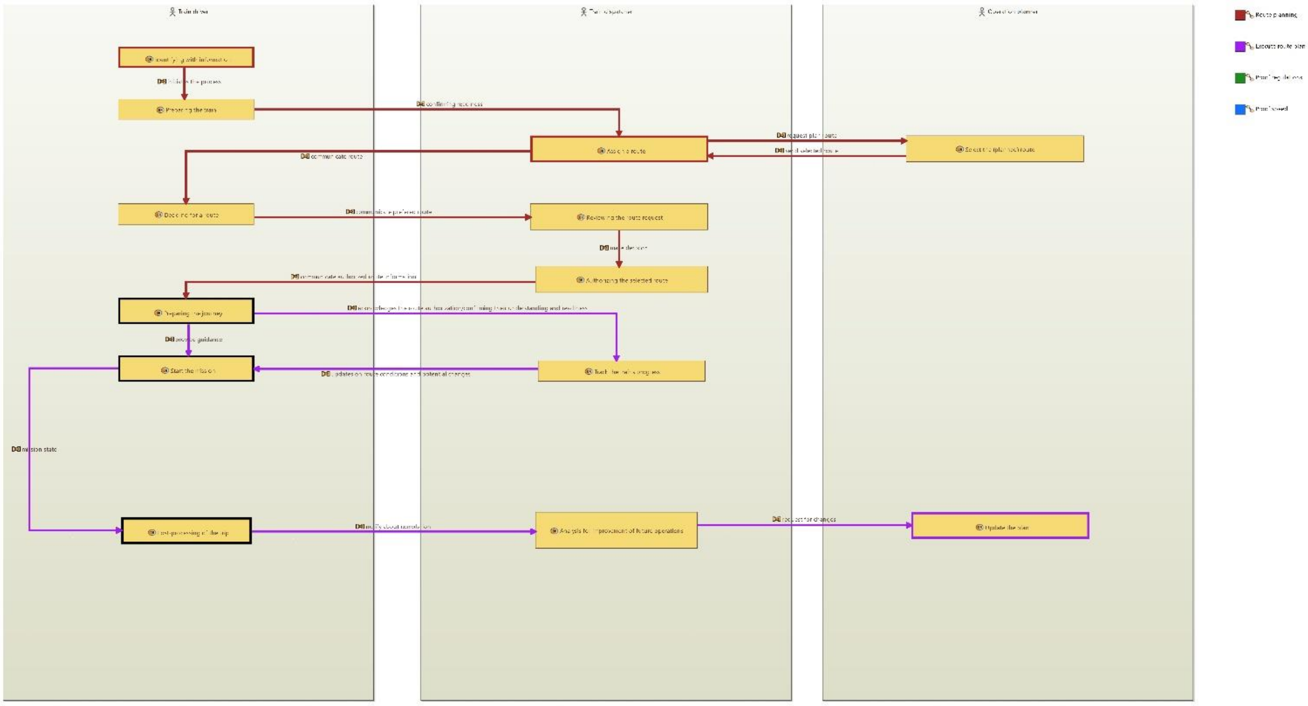

**Figure 20: Operational Architecture Diagram - Dynamic Route Changes**

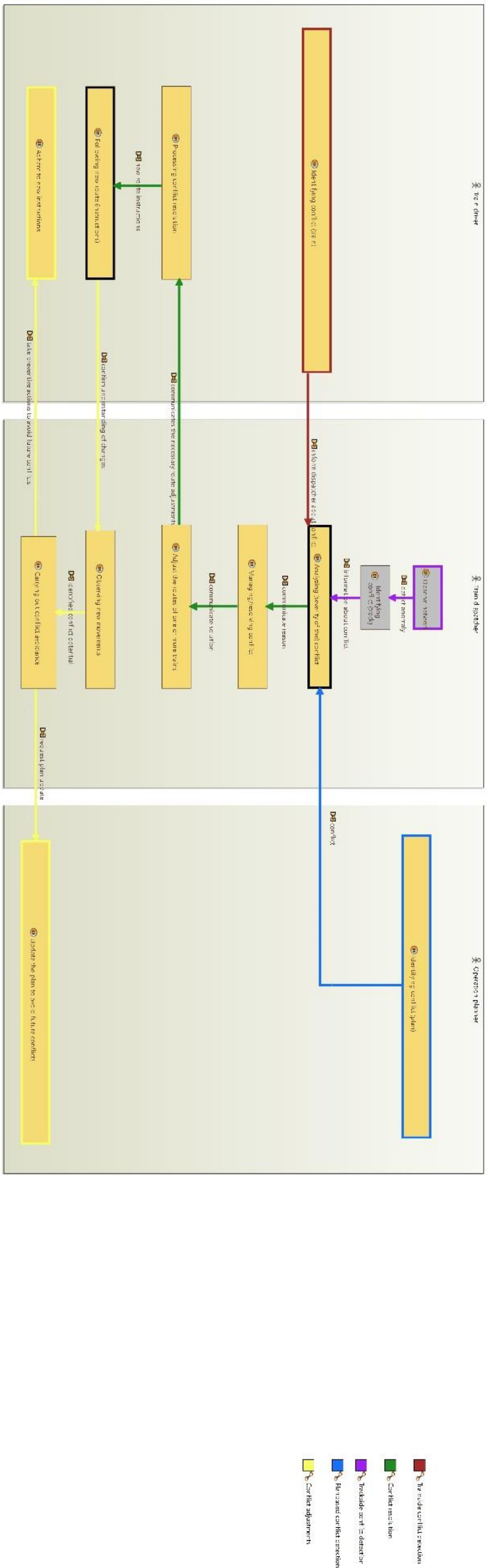

**Figure 21: Operational Architecture Diagram - Conflict Resolution**

## 4.4. FUNCTION SPECIFICATION - SYSTEM ANALYSIS[14]

The System Analysis of the ARCADIA method explores the functional and non-functional requirements of the system. It defines the system's behaviour, structure, and interfaces in response to operational needs, forming a bridge between operational goals and technical architecture.

### 4.4.1. Description of the System Missions

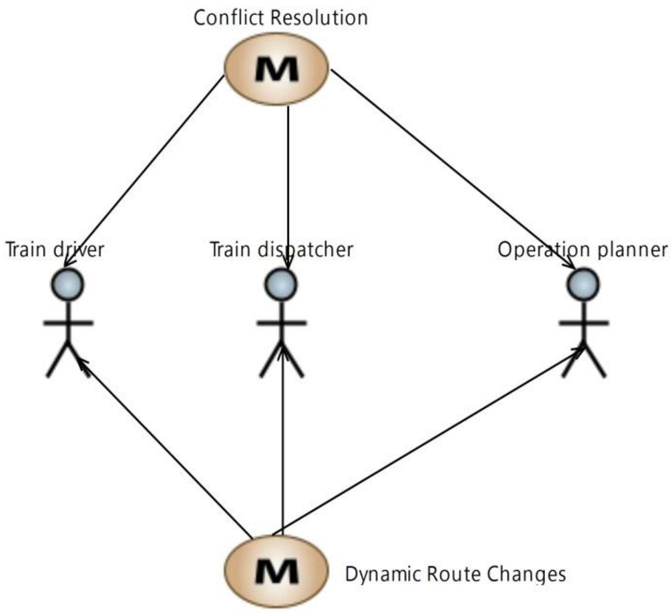


**Figure 22: Missions Diagram**

#### 4.4.1.1. Mission[15]: Dynamic Route Changes

Description:

dynamically adapt to changes in operational conditions, such as track maintenance, emergencies, or unexpected events

Involved actors:

- Train dispatcher
- Train driver

[14] generated automatically by M2Doc from Capella model of the system

[15] A purpose to which the System is tasked. A Mission generally exploits several System Capabilities.

- Operation planner

#### 4.4.1.1.1. Capability[16]: Routing decision

Description:

analyze real-time data and make routing decisions. System consider factors such as train schedules, priorities, track capacity, and potential conflicts to determine the most efficient routes.

Capability inclusion relations:

| Including capabilities | Current capability | Included capabilities |
|---|---|---|
| *No including capability* | Routing decision | *No included capability* |

Capability extension relations:

| Extended capabilities | Current capability | Extending capabilities |
|---|---|---|
| *No extended capability* | Routing decision | *No extending capability* |

Capability generalization relations:

| Super capabilities | Current capability | Sub capabilities |
|---|---|---|
| *No super capability* | Routing decision | *No sub capability* |

Involved functions:

*No involved function*

Involved functional chains:

*No involved functional chain*

Involved actors:

- Train dispatcher
- Train driver

[16] A Capability is the ability of the System to provide a service that supports the achievement of high-level operational goals. A Capability is described by scenarios and functional chains, all illustrating possible use cases. Capabilities can be used to structure the Functional Analysis. In Logical and Physical Architectures, Capabilities are called "Capability Realizations".

- Interlocking
- Signaller

Available in modes and states:

*No state or mode availability defined*

Involved scenarios:

- [ES] Routing decision

#### 4.4.1.1.2. Capability: Route adjustments

Description:

The system adjust the routes of one or more trains involved in the conflict. This could involve rerouting a train, delaying its departure, or holding it at a specific location until the conflict is resolved.

Capability inclusion relations:

| Including capabilities | Current capability | Included capabilities |
|---|---|---|
| *No including capability* | Route adjustments | *No included capability* |

Capability extension relations:

| Extended capabilities | Current capability | Extending capabilities |
|---|---|---|
| *No extended capability* | Route adjustments | *No extending capability* |

Capability generalization relations:

| Super capabilities | Current capability | Sub capabilities |
|---|---|---|
| *No super capability* | Route adjustments | *No sub capability* |

Involved functions:

*No involved function*

Involved functional chains:

*No involved functional chain*

Involved actors:

- Train driver
- Interlocking
- Infrastructure

Available in modes and states:

*No state or mode availability defined*

Involved scenarios:

- [ES] Route adjustments

##### 4.4.1.1.3. Capability: Route planning and optimization

Description:

system evaluates real-time data to dynamically plan and optimize train routes.

Capability inclusion relations:

| Including capabilities | Current capability | Included capabilities |
|---|---|---|
| *No including capability* | Route planning and optimization | *No included capability* |

Capability extension relations:

| Extended capabilities | Current capability | Extending capabilities |
|---|---|---|
| *No extended capability* | Route planning and optimization | *No extending capability* |

Capability generalization relations:

| Super capabilities | Current capability | Sub capabilities |
|---|---|---|
| *No super capability* | Route planning and | *No sub capability* |

| | | |
|---|---|---|
| | optimization | |

Involved functions:

*No involved function*

Involved functional chains:

*No involved functional chain*

Involved actors:

- Train driver
- Interlocking
- Train dispatcher
- Infrastructure

Available in modes and states:

*No state or mode availability defined*

Involved scenarios:

- [ES] Route optimization and planning

#### 4.4.1.2. Mission: Conflict Resolution

Description:

preventing conflicts between trains by intelligently selecting routes that avoid potential clashes. It considers factors like train speed, braking distances, and track occupancy to ensure safe and conflict-free operations.

Involved actors:

- Train driver
- Train dispatcher
- Operation planner

Exploited capabilities:

##### *4.4.1.2.1. Capability: Ensuring safe train movements*

Description:

Safety is paramount in dynamic routing systems. Route adjustments prioritize safety constraints and regulations, ensuring that trains are routed along paths that minimize the risk of collisions, derailments, or other safety hazards.

Capability inclusion relations:

| Including capabilities | Current capability | Included capabilities |
|---|---|---|
| *No including capability* | Ensuring safe train movements | *No included capability* |

Capability extension relations:

| Extended capabilities | Current capability | Extending capabilities |
|---|---|---|
| *No extended capability* | Ensuring safe train movements | *No extending capability* |

Capability generalization relations:

| Super capabilities | Current capability | Sub capabilities |
|---|---|---|
| *No super capability* | Ensuring safe train movements | *No sub capability* |

Involved functions:

*No involved function*

Involved functional chains:

*No involved functional chain*

Involved actors:

*No involved actor*

Available in modes and states:

*No state or mode availability defined*

Involved scenarios:

*No involved scenario*

#### 4.4.1.2.2. Capability: Detecting conflicts

Description:

The dynamic routing system detects conflicts between trains, such as overlapping routes or converging tracks. Algorithms analyze these conflicts and propose alternative routes to resolve them, ensuring that trains can proceed safely and without interruption.

Capability inclusion relations:

| Including capabilities | Current capability | Included capabilities |
|---|---|---|
| *No including capability* | Detecting conflicts | *No included capability* |

Capability extension relations:

| Extended capabilities | Current capability | Extending capabilities |
|---|---|---|
| *No extended capability* | Detecting conflicts | *No extending capability* |

Capability generalization relations:

| Super capabilities | Current capability | Sub capabilities |
|---|---|---|
| *No super capability* | Detecting conflicts | *No sub capability* |

Involved functions:

*No involved function*

Involved functional chains:

*No involved functional chain*

Involved actors:

- Infrastructure
- Interlocking
- Train driver
- Train dispatcher

Available in modes and states:

*No state or mode availability defined*

Involved scenarios:

- [ES] Detecting conflicts

##### 4.4.1.2.3. Capability: Conflict resolution and coordination

Description:

Dynamic routing capabilities include the ability to detect and resolve conflicts between trains in real-time. ARS algorithms can identify conflicts and automatically adjust routes to avoid collisions and ensure safe train movements.

Capability inclusion relations:

| **Including capabilities** | **Current capability** | **Included capabilities** |
| --- | --- | --- |
| *No including capability* | Conflict resolution and coordination | *No included capability* |

Capability extension relations:

| **Extended capabilities** | **Current capability** | **Extending capabilities** |
| --- | --- | --- |
| *No extended capability* | Conflict resolution and coordination | *No extending capability* |

Capability generalization relations:

| **Super capabilities** | **Current capability** | **Sub capabilities** |
| --- | --- | --- |
| *No super capability* | Conflict resolution and coordination | *No sub capability* |

Involved functions:

*No involved function*

Involved functional chains:

*No involved functional chain*

Involved actors:

*No involved actor*

Available in modes and states:

*No state or mode availability defined*

Involved scenarios:

- [ES] Conflict resolution and coordination

### 4.4.2. Description of the System Capabilities

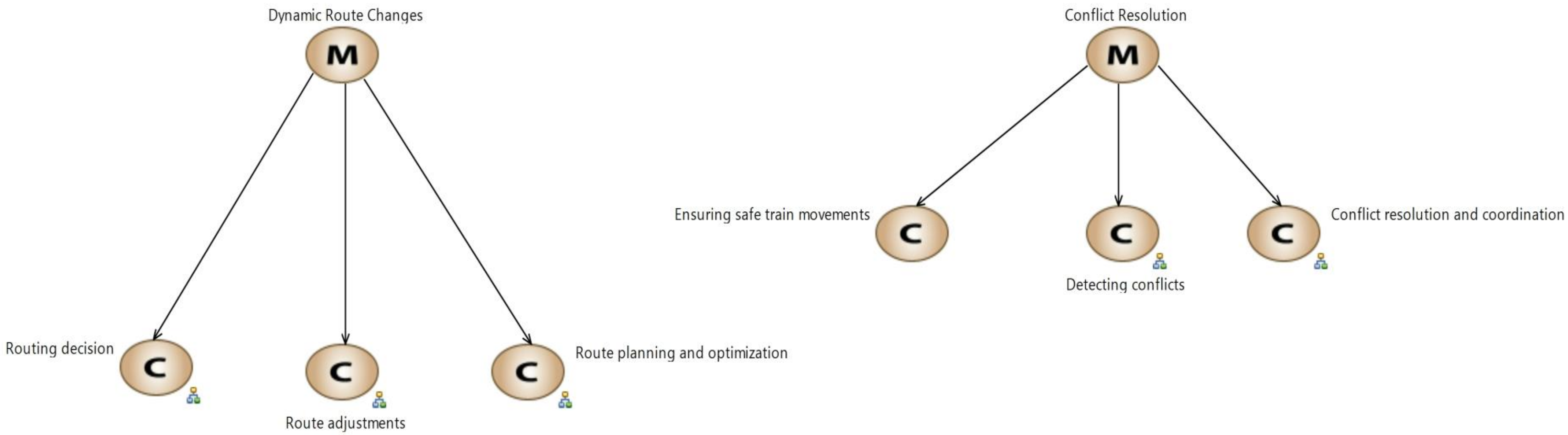


**Figure 23: Missions Capabilities Diagram**

### 4.4.3. Description of the System and its Environment

#### 4.4.3.1. Architecture Diagrams

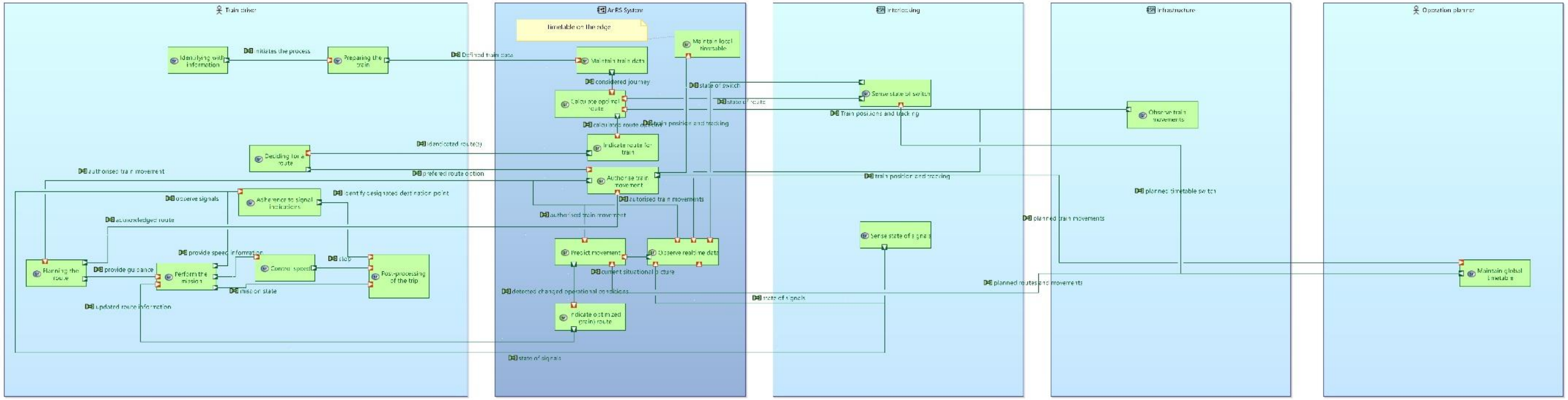

**Figure 24: System Architecture Diagram - Dynamic Route Changes**

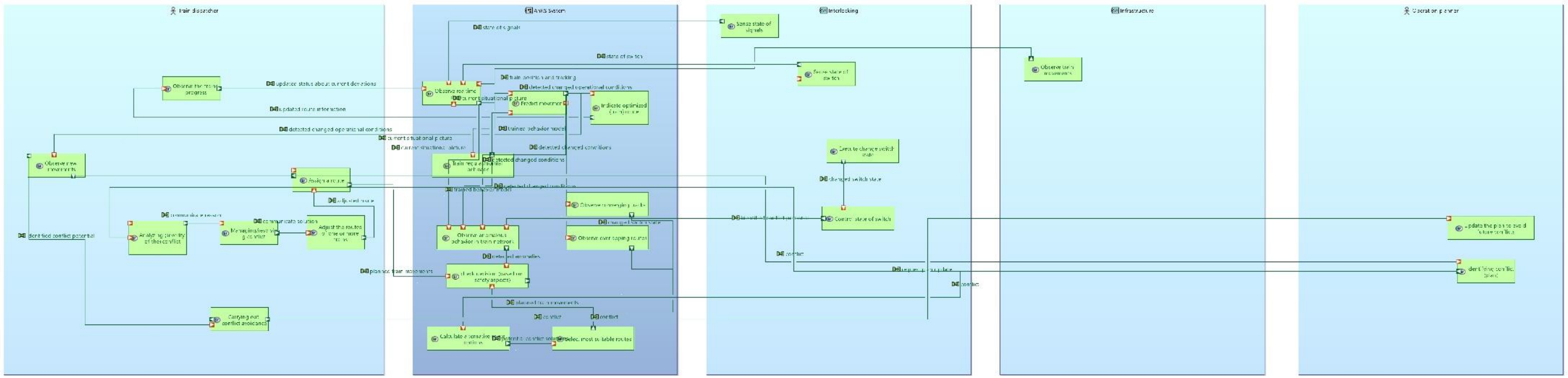


Figure 25: System Architecture Diagram - Conflict Resolution

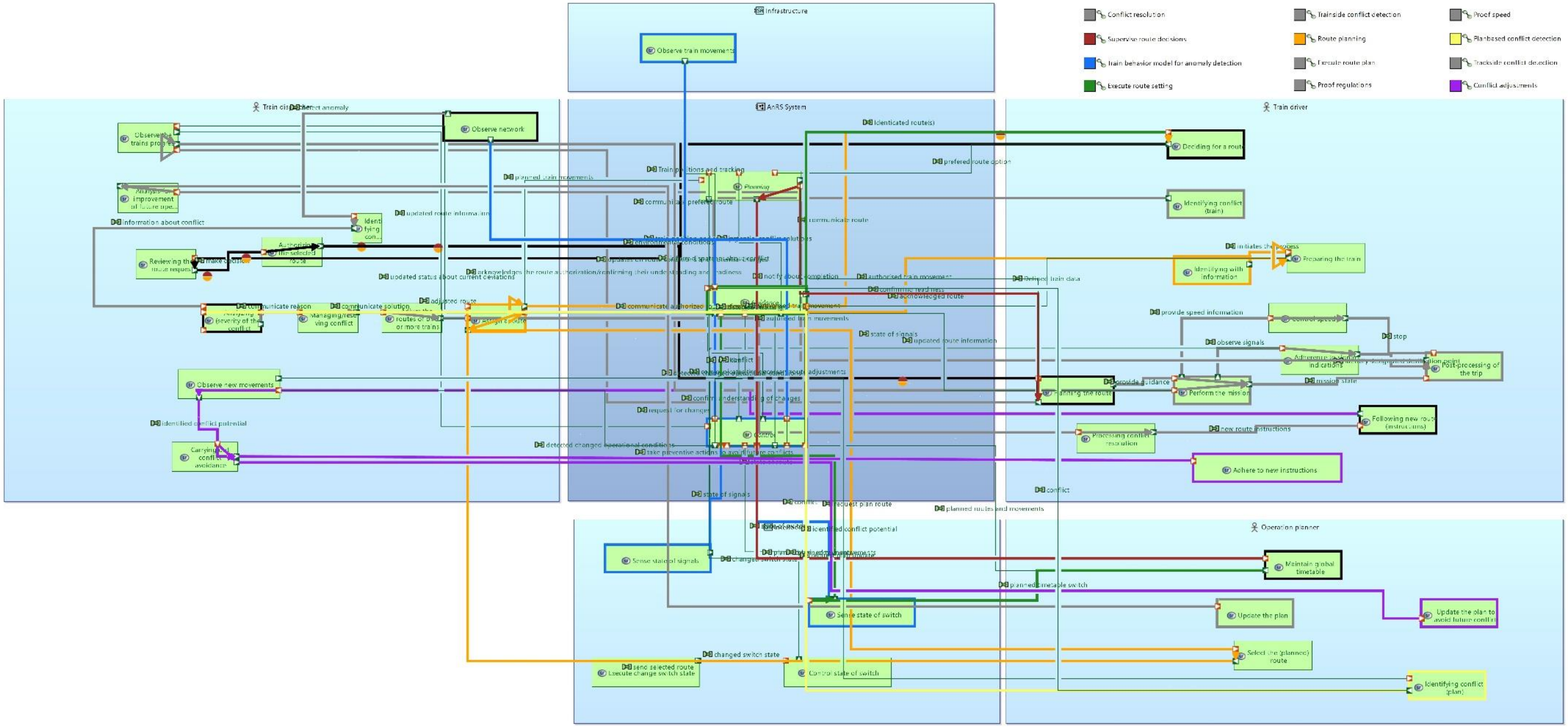


Figure 26: System Architecture Diagram - AnRS System - Structure

### 4.4.3.2. System: AnRS System

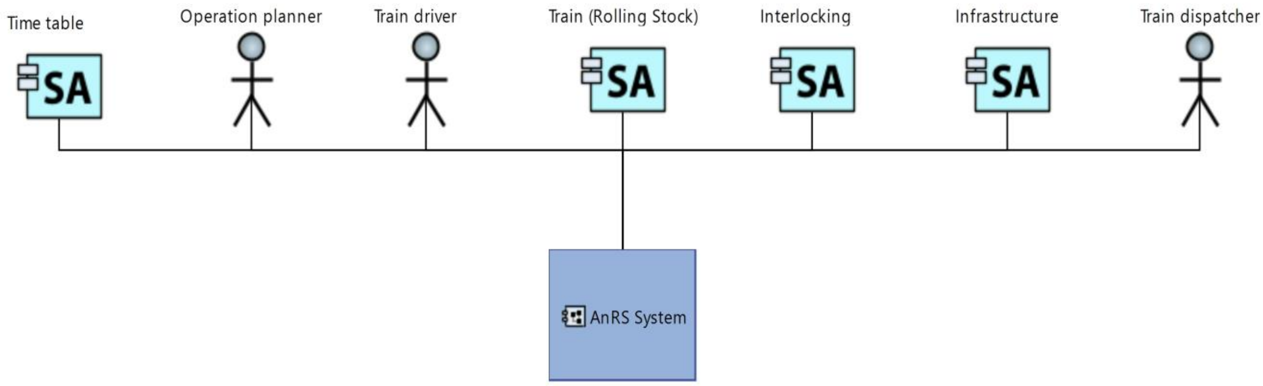


**Figure 27: Contextual System Actors Diagram**

Modes & State Machines:

- System State Machine

Allocated functions:

- Maintain train data
- Indicate route for train
- Authorise train movement
- Calculate optimal route
- Observe realtime data
- Indicate optimized (train) route
- Predict movement
- Observe anomalous behavior in train network
- Observe overlapping routes
- Observe converging tracks
- Calculate alternative route options
- Select most suitable routes
- Train regular/normal behavior
- check decision (based on safety aspects)
- Maintain local timetable

#### 4.4.3.3. Actor: Point machine

Modes & State Machines:

*No modes & states machine*

Allocated functions:

*No allocated function*

#### 4.4.3.4. Actor: Train dispatcher

Modes & State Machines:

*No modes & states machine*

Allocated functions:

- Assign a route
- Reviewing the route request
- Authorizing the selected route
- Observe the trains progress
- Analysis for improvement of future operations
- Managing/resolving conflict
- Analyzing (severity of the) conflict
- Adjust the routes of one or more trains
- Observe new movements
- Carrying out conflict avoidance
- Observe network
- Identifying conflict (track)

#### 4.4.3.5. Actor: Signal

Modes & State Machines:

*No modes & states machine*

Allocated functions:

*No allocated function*

#### 4.4.3.6. Actor: Switch

Modes & State Machines:

*No modes & states machine*

Allocated functions:

*No allocated function*

#### 4.4.3.7. Actor: Train driver

Modes & State Machines:

*No modes & states machine*

Allocated functions:

- Preparing the train
- Identifying with information
- Deciding for a route
- Planning the route
- Perform the mission
- Control speed
- Adherence to signal indications
- Post-processing of the trip
- Identifying conflict (train)
- Processing conflict resolution
- Following new route (instructions)
- Adhere to new instructions

#### 4.4.3.8. Actor: Train (Rolling Stock)

Modes & State Machines:

*No modes & states machine*

Allocated functions:

*No allocated function*

#### 4.4.3.9. Actor: Interlocking

Modes & State Machines:

*No modes & states machine*

Allocated functions:

- Sense state of switch
- Sense state of signals
- Control state of switch
- Execute change switch state

#### 4.4.3.10. Actor: Infrastructure

Modes & State Machines:

*No modes & states machine*

Allocated functions:

- Observe train movements

#### 4.4.3.11. Actor: Signaller

Modes & State Machines:

*No modes & states machine*

Allocated functions:

*No allocated function*

#### 4.4.3.12. Actor: Operation planner

Modes & State Machines:

*No modes & states machine*

Allocated functions:

- Update the plan
- Select the (planned) route
- Maintain global timetable
- Update the plan to avoid future conflicts
- Identifying conflict (plan)

#### 4.4.3.13. Actor: Time table

Modes & State Machines:

*No modes & states machine*

Allocated functions:

*No allocated function*

### 4.4.4. Description of the Functions

The functions and its hierarchy is shown in the following figure:

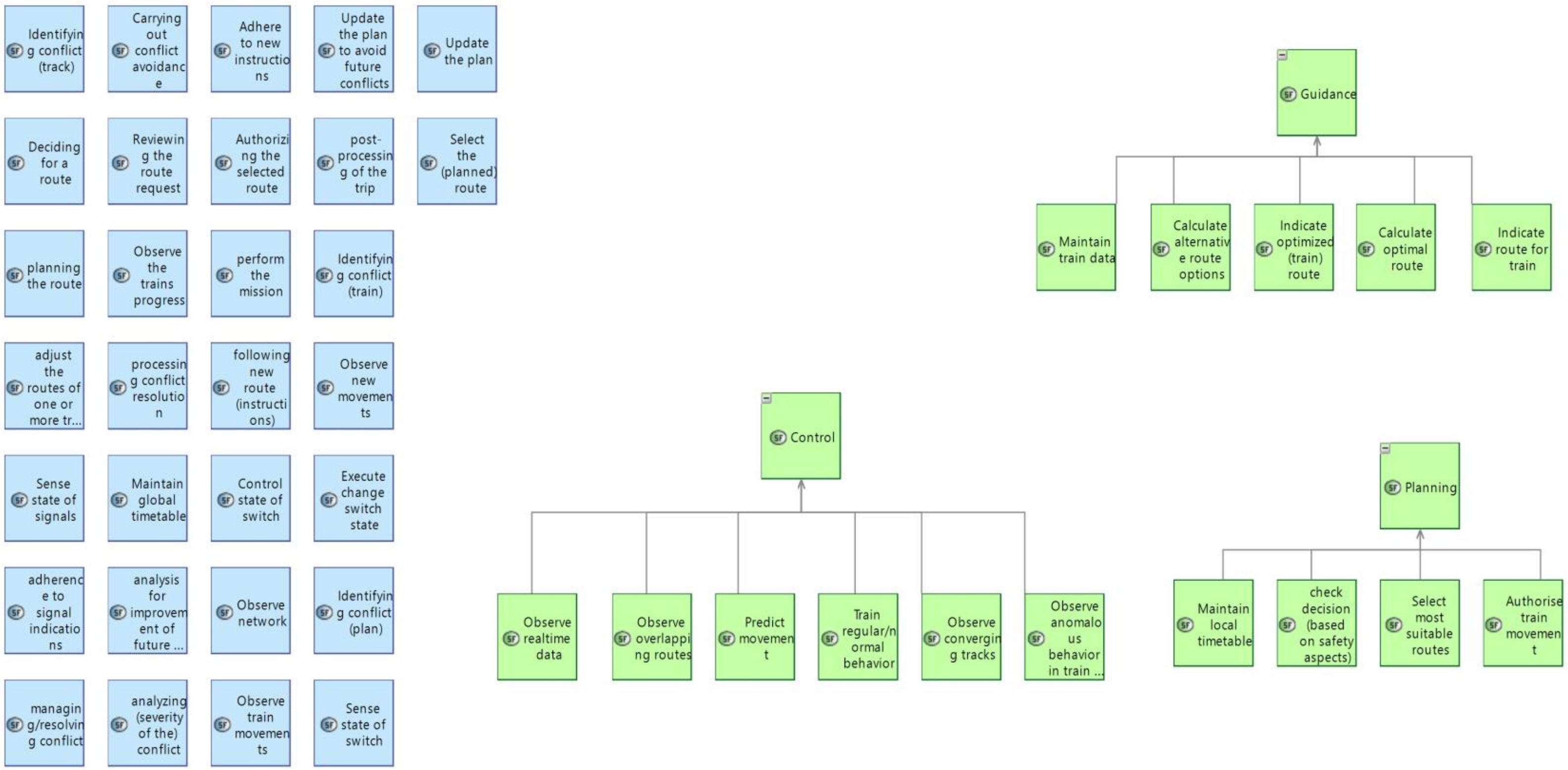


**Figure 28: System Function Breakdown Diagram**

The diagrams of the functions are shown below:

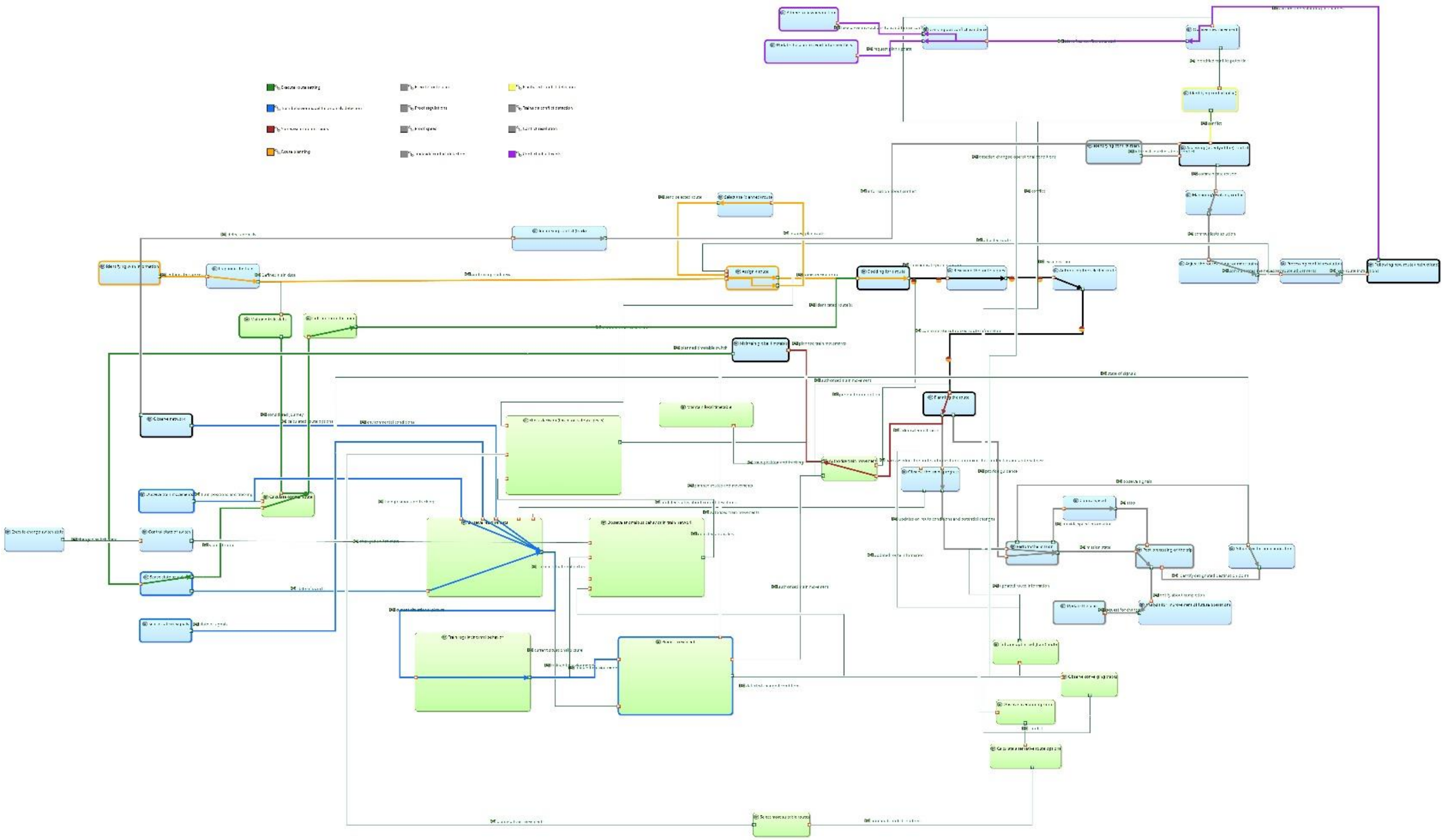

**Figure 29: System Data Flow Diagram**

The Functions are sorted in the following packages:

| Parent element | Function package | Contained functions |
|---|---|---|
| System Functions | Root System Function | Preparing the train<br>Identifying with information<br>Assign a route<br>Deciding for a route<br>Reviewing the route request<br>Authorizing the selected route<br>Planning the route<br>Observe the trains progress<br>Perform the mission<br>Control speed<br>Adherence to signal indications<br>Analysis for improvement of future operations<br>Post-processing of the trip<br>Observe network<br>Identifying conflict (track)<br>Identifying conflict (plan)<br>Identifying conflict (train)<br>Managing/resolving conflict<br>Analyzing (severity of the) conflict<br>Adjust the routes of one or more trains<br>Processing conflict resolution<br>Following new route |

| | | |
|---|---|---|
| | | (instructions)<br><br>Observe new movements<br><br>Observe train movements<br><br>Sense state of switch<br><br>Sense state of signals<br><br>Maintain global timetable<br><br>Control state of switch<br><br>Execute change switch state<br><br>Planning<br><br>Control<br><br>Guidance<br><br>Carrying out conflict avoidance<br><br>Adhere to new instructions<br><br>Update the plan to avoid future conflicts<br><br>Update the plan<br><br>Select the (planned) route |

#### 4.4.4.1. Function: Preparing the train

Description:

Rail operators create a schedule for trains, including departure and arrival times, and assign routes based on the overall network plan. Planning takes into account factors such as train priority, type of service (passenger or freight), and any scheduled maintenance on the tracks.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Preparing the train | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| initiates the process | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| confirming readiness | *No exchanged item* |
| Defined train data | *No exchanged item* |

Participation to functional chains[17]:

- Route planning

Allocated to:

- Train driver

#### 4.4.4.2. Function: Identifying with information

Description:

Each train is identified in the system, typically through a unique identifier or code. The train driver initiates the process by logging into the system, providing necessary identification details, and confirming the train's readiness.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Identifying with information | *No children function* |

List of inputs:

*No input*

List of outputs:

[17] Will be explained in section 4.4.5

| Outgoing functional exchange | Exchanged items |
|---|---|
| initiates the process | *No exchanged item* |

Participation to functional chains:

- Route planning

Allocated to:

- Train driver

#### 4.4.4.3. Function: Assign a route

Description:

Based on the schedule and network conditions, the dispatcher or central control system may automatically assign a route to the train.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Assign a route | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| confirming readiness | *No exchanged item* |
| adjusted route | *No exchanged item* |
| send selected route | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| communicate route | *No exchanged item* |
| planned train movements | *No exchanged item* |

| Outgoing functional exchange | Exchanged items |
| --- | --- |
| request plan route | *No exchanged item* |

Participation to functional chains:

- Route planning

Allocated to:

- Train dispatcher

#### 4.4.4.4. Function: Deciding for a route

Description:

The train driver may request a specific route if there are multiple options available.

Parent / Children functions:

| Parent function | Current function | Children functions |
| --- | --- | --- |
| Root System Function | Deciding for a route | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
| --- | --- |
| communicate route | *No exchanged item* |
| Identicated route(s) | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
| --- | --- |
| communicate prefered route | *No exchanged item* |
| prefered route option | *No exchanged item* |

Participation to functional chains:

- Supervise route decisions

- Execute route setting
- Route planning

Allocated to:

- Train driver

#### 4.4.4.5. Function: Reviewing the route request

Description:

The dispatcher reviews the route request, considering factors such as track availability, maintenance schedules, and other operational constraints. Once approved, the dispatcher authorizes the selected route for the train.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Reviewing the route request | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| communicate prefered route | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| make decision | *No exchanged item* |

Participation to functional chains:

- Supervise route decisions
- Route planning

Allocated to:

- Train dispatcher

#### 4.4.4.6. Function: Authorizing the selected route

Description:

The dispatcher communicates the authorized route information to the train driver. The train operator acknowledges the route authorization, confirming their understanding and readiness to proceed along the assigned route.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Authorizing the selected route | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| make decision | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| communicate authorized route information | *No exchanged item* |

Participation to functional chains:

- Supervise route decisions
- Route planning

Allocated to:

- Train dispatcher

#### 4.4.4.7. Function: Planning the route

Description:

The train's on-board control system receives the authorized route information and may provide guidance to the train operator. Automated train control systems may take over specific functions, such as speed control and adherence to signal indications.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Planning the route | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| communicate authorized route information | *No exchanged item* |
| authorised train movement | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| provide guidance | *No exchanged item* |
| acknowledges the route authorization/confirming their understanding and readiness | *No exchanged item* |
| acknowledged route | *No exchanged item* |

Participation to functional chains:

- Supervise route decisions
- Route planning
- Execute route plan

Allocated to:

- Train driver

#### 4.4.4.8. Function: Observe the trains progress

Description:

| |
|---|
| Both the dispatcher and the train operator have access to real-time monitoring systems. The dispatcher can track the train's progress, and the train operator can receive updates on route conditions and potential changes. |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Observe the trains progress | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| acknowledges the route authorization/confirming their understanding and readiness | *No exchanged item* |
| updated route information | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| updates on route conditions and potential changes | *No exchanged item* |
| updated status about current deviations | *No exchanged item* |

Participation to functional chains:

- Execute route plan

Allocated to:

- Train dispatcher

#### 4.4.4.9. Function: Perform the mission

Description:

| |
|---|
| The train driver starts the journey |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Perform the mission | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| provide guidance | *No exchanged item* |
| updates on route conditions and potential changes | *No exchanged item* |
| updated route information | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| provide speed information | *No exchanged item* |
| observe signals | *No exchanged item* |
| mission state | *No exchanged item* |

Participation to functional chains:

- Proof regulations
- Proof speed
- Execute route plan

Allocated to:

- Train driver

#### 4.4.4.10. Function: Control speed

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Control speed | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| provide speed information | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| stop | *No exchanged item* |

Participation to functional chains:

- Proof speed

Allocated to:

- Train driver

#### 4.4.4.11. Function: Adherence to signal indications

Description:

| |
|---|
| Signals along the track indicate to the train operator whether the route is clear, occupied, or if there are any speed restrictions. |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|

| | | |
|---|---|---|
| Root System Function | Adherence to signal indications | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| observe signals | *No exchanged item* |
| state of signals | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| identify designated destination point | *No exchanged item* |

Participation to functional chains:

- Proof regulations

Allocated to:

- Train driver

#### 4.4.4.12. Function: Analysis for improvement of future operations

Description:

| |
|---|
| After the conflict is resolved, the dispatcher and operational staff may conduct a post-resolution analysis. This involves reviewing the incident, understanding the root causes, and identifying any systemic improvements that can be made to enhance future conflict management. |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Analysis for improvement of future operations | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| notify about completion | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| request for changes | *No exchanged item* |

Participation to functional chains:

- Execute route plan

Allocated to:

- Train dispatcher

#### 4.4.4.13. Function: Post-processing of the trip

Description:

After the trip, the train operator may provide feedback on any issues encountered during the journey. The system provides feedback to operators on the status of trains, route changes, and any incidents. Reports and logs are generated for analysis and improvement of operational efficiency. The dispatcher uses this information for analysis and improvement of future operations.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Post-processing of the trip | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| identify designated destination point | *No exchanged item* |
| stop | *No exchanged item* |
| mission state | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
| --- | --- |
| notify about completion | *No exchanged item* |

Participation to functional chains:

- Proof regulations
- Proof speed
- Execute route plan

Allocated to:

- Train driver

#### 4.4.4.14. Function: Observe network

Description:

| |
| --- |
| various sensors, train positioning data, and real-time monitoring |

Parent / Children functions:

| Parent function | Current function | Children functions |
| --- | --- | --- |
| Root System Function | Observe network | *No children function* |

List of inputs:

*No input*

List of outputs:

| Outgoing functional exchange | Exchanged items |
| --- | --- |
| environmental conditions | *No exchanged item* |
| detect anomaly | *No exchanged item* |

Participation to functional chains:

- Train behavior model for anomaly detection
- Trackside conflict detection

Allocated to:

- Train dispatcher

#### 4.4.4.15. Function: Identifying conflict (track)

Description:

identifying conflicts based on the information available on the trackside.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Identifying conflict (track) | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| detect anomaly | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| information about conflict | *No exchanged item* |

Participation to functional chains:

- Trackside conflict detection

Allocated to:

- Train dispatcher

#### 4.4.4.16. Function: Identifying conflict (plan)

Description:

identifying conflicts based on the current and planned positions of trains

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Identifying conflict (plan) | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
| --- | --- |
| identified conflict potential | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
| --- | --- |
| conflict | *No exchanged item* |
| conflict | *No exchanged item* |

Participation to functional chains:

- Planbased conflict detection

Allocated to:

- Operation planner

#### 4.4.4.17. Function: Identifying conflict (train)

Description:

identifying conflicts based on the local information inside the train.

Parent / Children functions:

| Parent function | Current function | Children functions |
| --- | --- | --- |
| Root System Function | Identifying conflict (train) | *No children function* |

List of inputs:

*No input*

List of outputs:

| Outgoing functional exchange | Exchanged items |
| --- | --- |
| inform dispatcher about conflict | *No exchanged item* |

Participation to functional chains:

- Trainside conflict detection

Allocated to:

- Train driver

#### 4.4.4.18. Function: Managing/resolving conflict

Description:

| |
|---|
| The dynamic routing system detects conflicts between trains, such as overlapping routes or converging tracks. Algorithms analyze these conflicts and propose alternative routes to resolve them, ensuring that trains can proceed safely and without interruption. |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Managing/resolving conflict | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| communicate reason | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| communicate solution | *No exchanged item* |

Participation to functional chains:

- Conflict resolution

Allocated to:

- Train dispatcher

#### 4.4.4.19. Function: Analyzing (severity of the) conflict

Description:

| |
|---|
| The dispatcher analyzes the nature and severity of the conflict. This includes considering the speeds of the involved trains, the distance between them, and the type of track configuration contributing to the conflict. |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Analyzing (severity of the) conflict | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| inform dispatcher about conflict | *No exchanged item* |
| conflict | *No exchanged item* |
| information about conflict | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| communicate reason | *No exchanged item* |

Participation to functional chains:

- Trackside conflict detection
- Trainside conflict detection
- Conflict resolution
- Planbased conflict detection

Allocated to:

- Train dispatcher

#### 4.4.4.20. Function: Adjust the routes of one or more trains

Description:

The dispatcher may manually or automatically adjust the routes of one or more trains involved in the conflict. This could involve rerouting a train, delaying its departure, or holding it at a specific location until the conflict is resolved.

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Adjust the routes of one or more trains | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| communicate solution | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| communicates the necessary route adjustments | *No exchanged item* |
| adjusted route | *No exchanged item* |

Participation to functional chains:

- Conflict resolution

Allocated to:

- Train dispatcher

### 4.4.4.21. Function: Processing conflict resolution

Description:

| |
|---|
| The dispatcher communicates the necessary route adjustments to the affected train operators. This communication is crucial to ensuring that train operators are aware of the changes and can adhere to the new route instructions. |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Processing conflict resolution | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| communicates the necessary route adjustments | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| new route instructions | *No exchanged item* |

Participation to functional chains:

- Conflict resolution

Allocated to:

- Train driver

#### 4.4.4.22. Function: Following new route (instructions)

Description:

| |
|---|
| The train driver follows the new route |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Following new route (instructions) | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| new route instructions | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| confirm understanding of changes | *No exchanged item* |

Participation to functional chains:

- Conflict resolution
- Conflict adjustments

Allocated to:

- Train driver

#### 4.4.4.23. Function: Observe new movements

Description:

Throughout the conflict resolution process, the central control system continues to monitor the updated positions and movements of the trains involved. This ongoing monitoring helps ensure that the conflict is effectively resolved.

Parent / Children functions:

| **Parent function** | **Current function** | **Children functions** |
|---|---|---|
| Root System Function | Observe new movements | *No children function* |

List of inputs:

| **Incoming functional exchange** | **Exchanged items** |
|---|---|
| confirm understanding of changes | *No exchanged item* |
| detected changed operational conditions | *No exchanged item* |

List of outputs:

| **Outgoing functional exchange** | **Exchanged items** |
|---|---|
| identified conflict potential | *No exchanged item* |
| identified conflict potential | *No exchanged item* |

Participation to functional chains:

- Conflict adjustments

Allocated to:

- Train dispatcher

#### 4.4.4.24. Function: Observe train movements

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Observe train movements | *No children function* |

List of inputs:

*No input*

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| Train positions and tracking | *No exchanged item* |
| train position and tracking | *No exchanged item* |

Participation to functional chains:

- Train behavior model for anomaly detection

Allocated to:

- Infrastructure

#### 4.4.4.25. Function: Sense state of switch

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|

| | | |
|---|---|---|
| Root System Function | Sense state of switch | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| planned timetable switch | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| state of route | *No exchanged item* |
| state of switch | *No exchanged item* |

Participation to functional chains:

- Train behavior model for anomaly detection
- Execute route setting

Allocated to:

- Interlocking

#### 4.4.4.26. Function: Sense state of signals

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Sense state of signals | *No children function* |

List of inputs:

*No input*

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| state of signals | *No exchanged item* |
| state of signals | *No exchanged item* |

Participation to functional chains:

- Train behavior model for anomaly detection

Allocated to:

- Interlocking

#### 4.4.4.27. Function: Maintain global timetable

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Maintain global timetable | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| planned train movements | *No exchanged item* |
| planned train movements | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| planned routes and movements | *No exchanged item* |
| planned timetable switch | *No exchanged item* |

Participation to functional chains:

- Supervise route decisions

- Execute route setting

Allocated to:

- Operation planner

#### 4.4.4.28. Function: Control state of switch

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Control state of switch | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| changed switch state | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| changed switch state | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- Interlocking

#### 4.4.4.29. Function: Execute change switch state

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|

| | | |
|---|---|---|
| Root System Function | Execute change switch state | *No children function* |

List of inputs:

*No input*

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| changed switch state | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- Interlocking

#### 4.4.4.30. Function: Planning

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Planning | • check decision (based on safety aspects)<br>• Select most suitable routes<br>• Authorise train movement<br>• Maintain local timetable |

List of inputs:

*No input*

List of outputs:

*No output*

Participation to functional chains:

*None*

Allocated to:

*Function not allocated*

#### 4.4.4.31. Function: check decision (based on safety aspects)

Description:

| |
|---|
| Evaluate/Supervise decision |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Planning | check decision (based on safety aspects) | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| planned train movements | *No exchanged item* |
| detected anomalies | *No exchanged item* |
| planned train movements | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| planned train movements | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.32. Function: Select most suitable routes

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Planning | Select most suitable routes | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| potential conflict solutions | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| planned train movements | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.33. Function: Authorise train movement

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Planning | Authorise train movement | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| prefered route option | *No exchanged item* |
| acknowledged route | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| authorised train movement | *No exchanged item* |
| authorised train movement | *No exchanged item* |
| autorised train movements | *No exchanged item* |
| train position and tracking | *No exchanged item* |
| planned train movements | *No exchanged item* |

Participation to functional chains:

- Supervise route decisions

Allocated to:

- AnRS System

#### 4.4.4.34. Function: Maintain local timetable

Description:

| |
|---|
| has to be considered |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Planning | Maintain local timetable | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| train position and tracking | *No exchanged item* |

List of outputs:

*No output*

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.35. Function: Control

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Control | • Observe overlapping routes<br>• Predict movement<br>• Train regular/normal behavior<br>• Observe converging tracks<br>• Observe anomalous behavior in train network<br>• Observe realtime data |

List of inputs:

*No input*

List of outputs:

*No output*

Participation to functional chains:

*None*

Allocated to:

*Function not allocated*

### 4.4.4.36. Function: Observe overlapping routes

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Control | Observe overlapping routes | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| detected changed conditions | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| conflict | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.37. Function: Predict movement

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Control | Predict movement | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| authorised train movement | *No exchanged item* |
| trained behavior model | *No exchanged item* |
| planned routes and movements | *No exchanged item* |
| current situational picture | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| detected changed conditions | *No exchanged item* |
| detected changed conditions | *No exchanged item* |
| detected changed conditions | *No exchanged item* |
| detected changed operational conditions | *No exchanged item* |
| detected changed operational conditions | *No exchanged item* |

Participation to functional chains:

- Train behavior model for anomaly detection

Allocated to:

- AnRS System

### 4.4.4.38. Function: Train regular/normal behavior

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Control | Train regular/normal behavior | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| current situational picture | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| trained behavior model | *No exchanged item* |
| trained behavior model | *No exchanged item* |

Participation to functional chains:

- Train behavior model for anomaly detection

Allocated to:

- AnRS System

### 4.4.4.39. Function: Observe converging tracks

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Control | Observe converging tracks | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| detected changed conditions | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| conflict | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.40. Function: Observe anomalous behavior in train network

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Control | Observe anomalous behavior in train network | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| trained behavior model | *No exchanged item* |
| detected changed conditions | *No exchanged item* |
| current situational picture | *No exchanged item* |
| changed switch state | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| detected anomalies | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.41. Function: Observe realtime data

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Control | Observe realtime data | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| train position and tracking | *No exchanged item* |
| autorised train movements | *No exchanged item* |
| updated status about current deviations | *No exchanged item* |
| state of signals | *No exchanged item* |
| state of switch | *No exchanged item* |
| environmental conditions | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| current situational picture | *No exchanged item* |

| Outgoing functional exchange | Exchanged items |
|---|---|
| current situational picture | *No exchanged item* |
| current situational picture | *No exchanged item* |

Participation to functional chains:

- Train behavior model for anomaly detection

Allocated to:

- AnRS System

#### 4.4.4.42. Function: Guidance

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Guidance | • Calculate alternative route options<br>• Indicate optimized (train) route<br>• Calculate optimal route<br>• Indicate route for train<br>• Maintain train data |

List of inputs:

*No input*

List of outputs:

*No output*

Participation to functional chains:

*None*

Allocated to:

*Function not allocated*

#### 4.4.4.43. Function: Calculate alternative route options

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Guidance | Calculate alternative route options | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| conflict | *No exchanged item* |
| conflict | *No exchanged item* |
| conflict | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| potential conflict solutions | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.44. Function: Indicate optimized (train) route

Description:

| |
|---|
| Conflict resolution |

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|

| | | |
|---|---|---|
| Guidance | Indicate optimized (train) route | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| detected changed operational conditions | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| updated route information | *No exchanged item* |
| updated route information | *No exchanged item* |

Participation to functional chains:

*None*

Allocated to:

- AnRS System

#### 4.4.4.45. Function: Calculate optimal route

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Guidance | Calculate optimal route | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| considered journey | *No exchanged item* |
| state of route | *No exchanged item* |
| Train positions and tracking | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| calculated route options | *No exchanged item* |

Participation to functional chains:

- Execute route setting

Allocated to:

- AnRS System

### 4.4.4.46. Function: Indicate route for train

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Guidance | Indicate route for train | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| calculated route options | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| Identicated route(s) | *No exchanged item* |

Participation to functional chains:

- Execute route setting

Allocated to:

- AnRS System

### 4.4.4.47. Function: Maintain train data

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Guidance | Maintain train data | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
|---|---|
| Defined train data | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| considered journey | *No exchanged item* |

Participation to functional chains:

- Execute route setting

Allocated to:

- AnRS System

### 4.4.4.48. Function: Carrying out conflict avoidance

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
|---|---|---|
| Root System Function | Carrying out conflict avoidance | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
| --- | --- |
| identified conflict potential | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
| --- | --- |
| take preventive actions to avoid future conflicts | *No exchanged item* |
| request plan update | *No exchanged item* |
| Interaction 31 | *No exchanged item* |
| Interaction 32 | *No exchanged item* |

Participation to functional chains:

- Conflict adjustments

Allocated to:

- Train dispatcher

#### 4.4.4.49. Function: Adhere to new instructions

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
| --- | --- | --- |
| Root System Function | Adhere to new instructions | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
| --- | --- |
| take preventive actions to avoid future conflicts | *No exchanged item* |

List of outputs:

*No output*

Participation to functional chains:

- Conflict adjustments

Allocated to:

- Train driver

#### 4.4.4.50. Function: Update the plan to avoid future conflicts

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
| --- | --- | --- |
| Root System Function | Update the plan to avoid future conflicts | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
| --- | --- |
| request plan update | *No exchanged item* |
| Interaction 31 | *No exchanged item* |
| Interaction 32 | *No exchanged item* |

List of outputs:

*No output*

Participation to functional chains:

- Conflict adjustments

Allocated to:

- Operation planner

#### 4.4.4.51. Function: Update the plan

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
| --- | --- | --- |
| Root System Function | Update the plan | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
| --- | --- |
| request for changes | *No exchanged item* |

List of outputs:

*No output*

Participation to functional chains:

- Execute route plan

Allocated to:

- Operation planner

#### 4.4.4.52. Function: Select the (planned) route

Description:

*No description*

Parent / Children functions:

| Parent function | Current function | Children functions |
| --- | --- | --- |
| Root System Function | Select the (planned) route | *No children function* |

List of inputs:

| Incoming functional exchange | Exchanged items |
| --- | --- |
| request plan route | *No exchanged item* |

List of outputs:

| Outgoing functional exchange | Exchanged items |
|---|---|
| send selected route | *No exchanged item* |

Participation to functional chains:

- Route planning

Allocated to:

- Operation planner

### 4.4.5. Description of the Functional Chains[18]

#### 4.4.5.1. Functional Chain: Train behavior model for anomaly detection

Description:

Train the behavior to detect anomalies.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| current situational picture | Observe realtime data | Train regular/normal behavior |
| environmental conditions | Observe network | Observe realtime data |
| state of signals | Sense state of signals | Observe realtime data |
| train position and tracking | Observe train movements | Observe realtime data |
| state of switch | Sense state of switch | Observe realtime data |
| trained behavior model | Train regular/normal behavior | Predict movement |

Involving capabilities:

*No involving capability*

[18] A Functional Chain is a mean to describe one specific path among all possible paths traversing the dataflow •either to describe an expected behavior of the System in a given context, or in order to express some non functional properties along this functional path (latency, criticality, confidentiality, redundancy...). An elementary Functional Chain is made of a succession of Functions and Exchanges across a functional dataflow, starting and ending with a Function (allocated to the System or to an Actor).

#### 4.4.5.2. Functional Chain: Supervise route decisions

Description:

The system provides feedback on route adjustments and monitors the effectiveness of routing decisions. This feedback loop allows for continuous improvement and optimization of train movements within the network.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| communicate authorized route information | Authorizing the selected route | Planning the route |
| acknowledged route | Planning the route | Authorise train movement |
| make decision | Reviewing the route request | Authorizing the selected route |
| communicate prefered route | Deciding for a route | Reviewing the route request |
| planned train movements | Authorise train movement | Maintain global timetable |

Involving capabilities:

*No involving capability*

#### 4.4.5.3. Functional Chain: Execute route setting

Description:

*No description*

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| planned timetable switch | Maintain global timetable | Sense state of switch |
| state of route | Sense state of switch | Calculate optimal route |
| considered journey | Maintain train data | Calculate optimal route |
| calculated route options | Calculate optimal route | Indicate route for train |
| Identicated route(s) | Indicate route for train | Deciding for a route |

Involving capabilities:

*No involving capability*

#### 4.4.5.4. Functional Chain: Trackside conflict detection

Description:

The trackside conflict detection shows the interaction between the relevant entities to make the detection of anomalies or conflicts available on the AnRS side. This is just the perspective from the track wich is equipped with various sensors, train positioning data, and real-time monitoring tools, detects potential conflicts. Conflicts may arise due to overlapping routes, converging tracks, or other factors. The central control system alerts the dispatcher about the detected conflict. The dispatcher is a key actor responsible for managing and resolving conflicts in real-time.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| detect anomaly | Observe network | Identifying conflict (track) |
| information about conflict | Identifying conflict (track) | Analyzing (severity of the) conflict |

Involving capabilities:

*No involving capability*

#### 4.4.5.5. Functional Chain: Trainside conflict detection

Description:

The trainside conflict detection shows the interaction between the relevant entities to make the detection of anomalies or conflicts available on the AnRS side. This is just the perspective from the train which is equipped with various sensors, train positioning data, and real-time monitoring tools, detects potential conflicts. Conflicts may arise due to overlapping routes, converging tracks, or other factors. The central control system alerts the dispatcher about the detected conflict. The dispatcher is a key actor responsible for managing and resolving conflicts in real-time.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| inform dispatcher about conflict | Identifying conflict (train) | Analyzing (severity of the) conflict |

Involving capabilities:

*No involving capability*

#### 4.4.5.6. Functional Chain: Conflict resolution

Description:

The dispatcher analyzes the nature and severity of the conflict. This includes considering the speeds of the involved trains, the distance between them, and the type of track configuration contributing to the conflict. The dispatcher communicates the necessary route adjustments to the affected train operators. This communication is crucial to ensuring that train operators are aware of the changes and can adhere to the new route instructions. Train operators acknowledge the route adjustments and confirm their understanding of the changes. This acknowledgment ensures that the train operators are aware of the new instructions and are ready to follow them.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| communicate reason | Analyzing (severity of the) conflict | Managing/resolving conflict |
| communicate solution | Managing/resolving conflict | Adjust the routes of one or more trains |
| communicates the necessary route adjustments | Adjust the routes of one or more trains | Processing conflict resolution |
| new route instructions | Processing conflict resolution | Following new route (instructions) |

Involving capabilities:

*No involving capability*

#### 4.4.5.7. Functional Chain: Route planning

Description:

The route planning describes how the route planning will be performed on a strategic level.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| initiates the process | Identifying with information | Preparing the train |
| confirming readiness | Preparing the train | Assign a route |
| communicate route | Assign a route | Deciding for a route |
| communicate preferred route | Deciding for a route | Reviewing the route request |
| make decision | Reviewing the route request | Authorizing the selected route |
| communicate authorized route information | Authorizing the selected route | Planning the route |
| request plan route | Assign a route | Select the (planned) route |
| send selected route | Select the (planned) route | Assign a route |

Involving capabilities:

*No involving capability*

#### 4.4.5.8. Functional Chain: Proof regulations

Description:

The assistant systems observes/monitor the mission movement and check against local regulations to guarantee compliance

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| observe signals | Perform the mission | Adherence to signal indications |
| identify designated destination point | Adherence to signal indications | Post-processing of the trip |

Involving capabilities:

*No involving capability*

#### 4.4.5.9. Functional Chain: Proof speed

Description:

The assistant systems observes/monitor the mission movement and check against speed restrictions/limitations to guarantee compliance

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| provide speed information | Perform the mission | Control speed |
| stop | Control speed | Post-processing of the trip |

Involving capabilities:

*No involving capability*

#### 4.4.5.10. Functional Chain: Planbased conflict detection

Description:

The OCB diagram shows the general capability from operational side that has to be considered by the AnRS. The overall goal of the rail domain is in general the optimization of the train flows, without safety reduction and realistic capacities. the main involved actors in here are the train drivers as well as the train dispatchers, so the capabilities that the AnRS directly can address to optimize the train flow is the dynamic route change and automatic conflict resolution. that's why we will focus on describing these operational processes for the Operational and System Analysis.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| conflict | Identifying conflict (plan) | Analyzing (severity of the) conflict |

Involving capabilities:

*No involving capability*

#### 4.4.5.11. Functional Chain: Conflict adjustments

Description:

The dispatcher may manually or automatically adjust the routes of one or more trains involved in the conflict. This could involve rerouting a train, delaying its departure, or holding it at a specific location until the conflict is resolved. In addition to immediate conflict resolution, the dispatcher may take preventive actions to avoid future conflicts. This could involve adjusting schedules, optimizing routes, or implementing other operational measures to minimize the likelihood of conflicts.

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| confirm understanding of changes | Following new route (instructions) | Observe new movements |
| identified conflict potential | Observe new movements | Carrying out conflict avoidance |
| request plan update | Carrying out conflict avoidance | Update the plan to avoid future conflicts |
| take preventive actions to avoid future conflicts | Carrying out conflict avoidance | Adhere to new instructions |

Involving capabilities:

*No involving capability*

#### 4.4.5.12. Functional Chain: Execute route plan

Description:

The train is ready to move through the defined route plan from beginning to end

Involved elements:

This functional chain involves the following elements:

| Functional exchange | Source function | Target function |
|---|---|---|
| provide guidance | Planning the route | Perform the mission |
| acknowledges the route authorization/confirming their understanding and readiness | Planning the route | Observe the trains progress |
| updates on route conditions and potential changes | Observe the trains progress | Perform the mission |
| mission state | Perform the mission | Post-processing of the trip |
| notify about completion | Post-processing of the trip | Analysis for improvement of future operations |

| Functional exchange | Source function | Target function |
|---|---|---|
| request for changes | Analysis for improvement of future operations | Update the plan |

Involving capabilities:

*No involving capability*

### 4.4.6. Description of Scenarios

#### 4.4.6.1. Scenario: [ES] Routing decision

Description:

*No description*

Allocated capability:

- Routing decision

Representation:

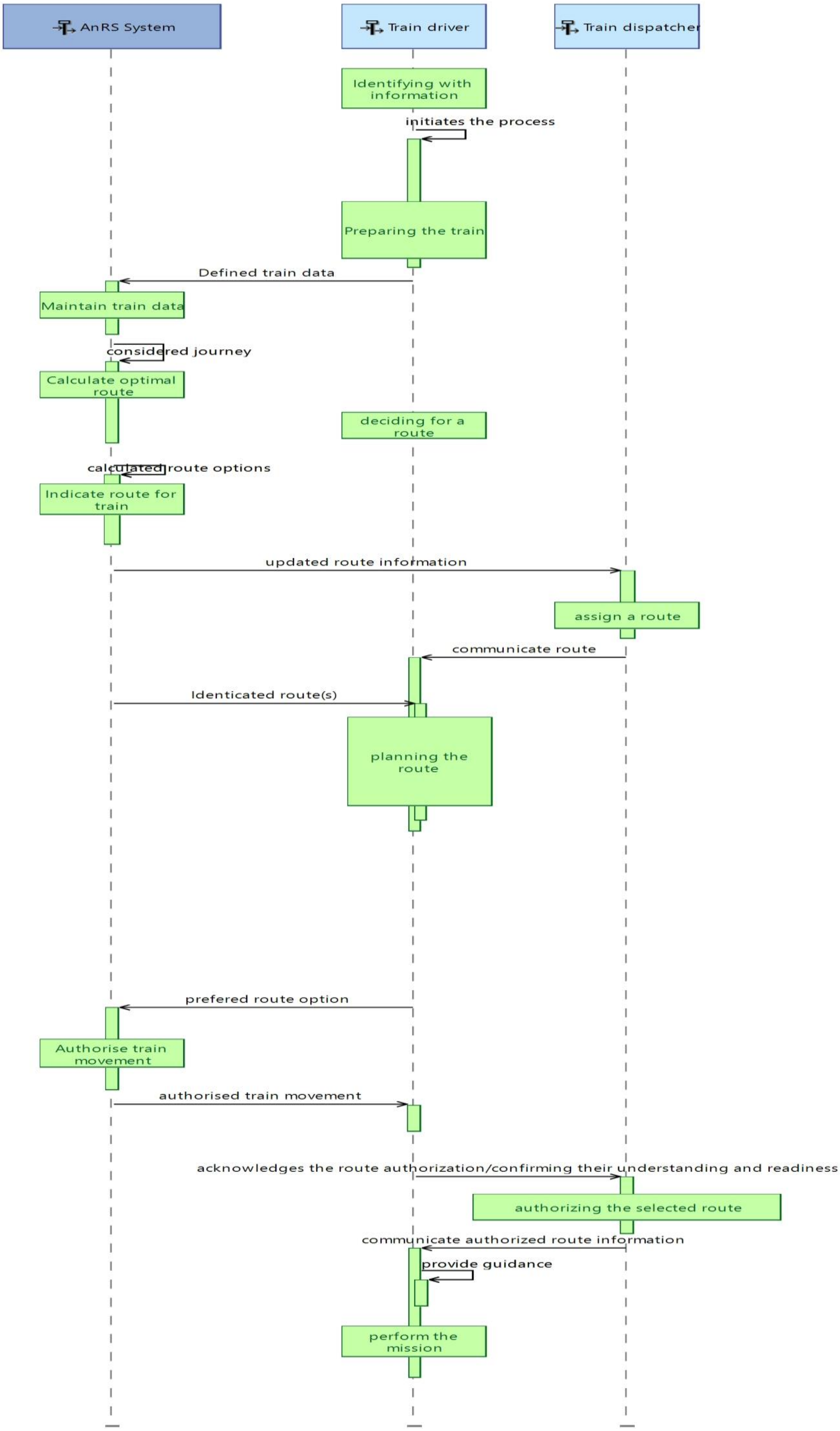


**Figure 30: Exchange Scenario Diagram**

Owned messages:

- initiates the process
- Defined train data
- considered journey
- calculated route options
- updated route information
- communicate route
- Identicated route(s)
- prefered route option
- authorised train movement
- acknowledges the route authorization/confirming their understanding and readiness
- communicate authorized route information
- provide guidance

#### 4.4.6.2. Scenario: [ES] Detecting conflicts

Description:

*No description*

Allocated capability:

- Detecting conflicts

Representation:

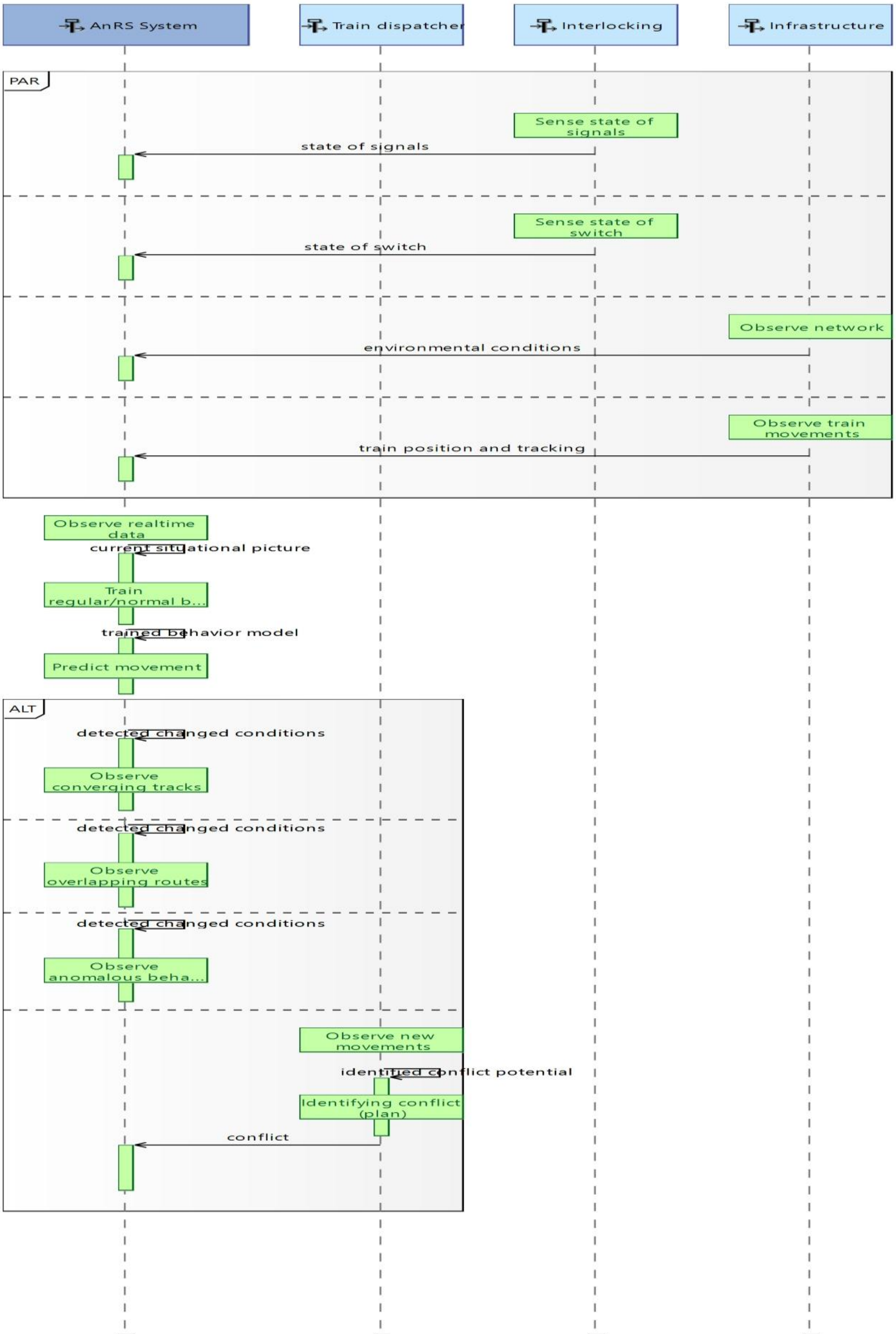


**Figure 31: Exchange Scenario Diagram**

Owned messages:

- state of signals
- state of switch
- environmental conditions
- train position and tracking
- current situational picture
- trained behavior model
- detected changed conditions
- detected changed conditions
- detected changed conditions
- identified conflict potential
- conflict

#### 4.4.6.3. Scenario: [ES] Route optimization and planning

Description:

*No description*

Allocated capability:

- Route planning and optimization

Representation:

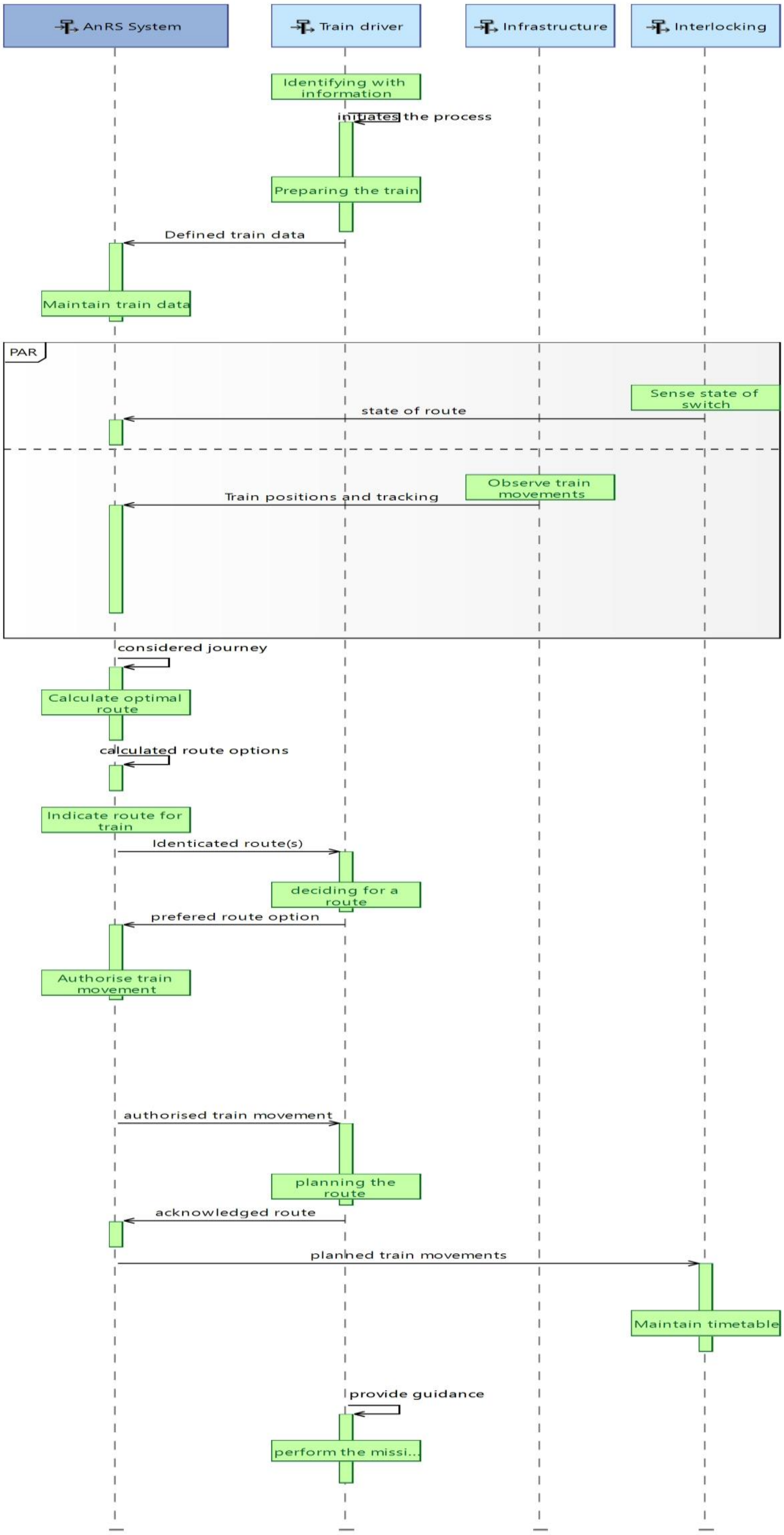


**Figure 32: Exchange Scenario Diagram**

Owned messages:

- initiates the process
- Defined train data
- state of route
- Train positions and tracking
- considered journey
- calculated route options
- Identicated route(s)
- prefered route option
- authorised train movement
- acknowledged route
- planned train movements
- provide guidance

#### 4.4.6.4. Scenario: [ES] Route adjustments

Description:

*No description*

Allocated capability:

- Route adjustments

Representation:

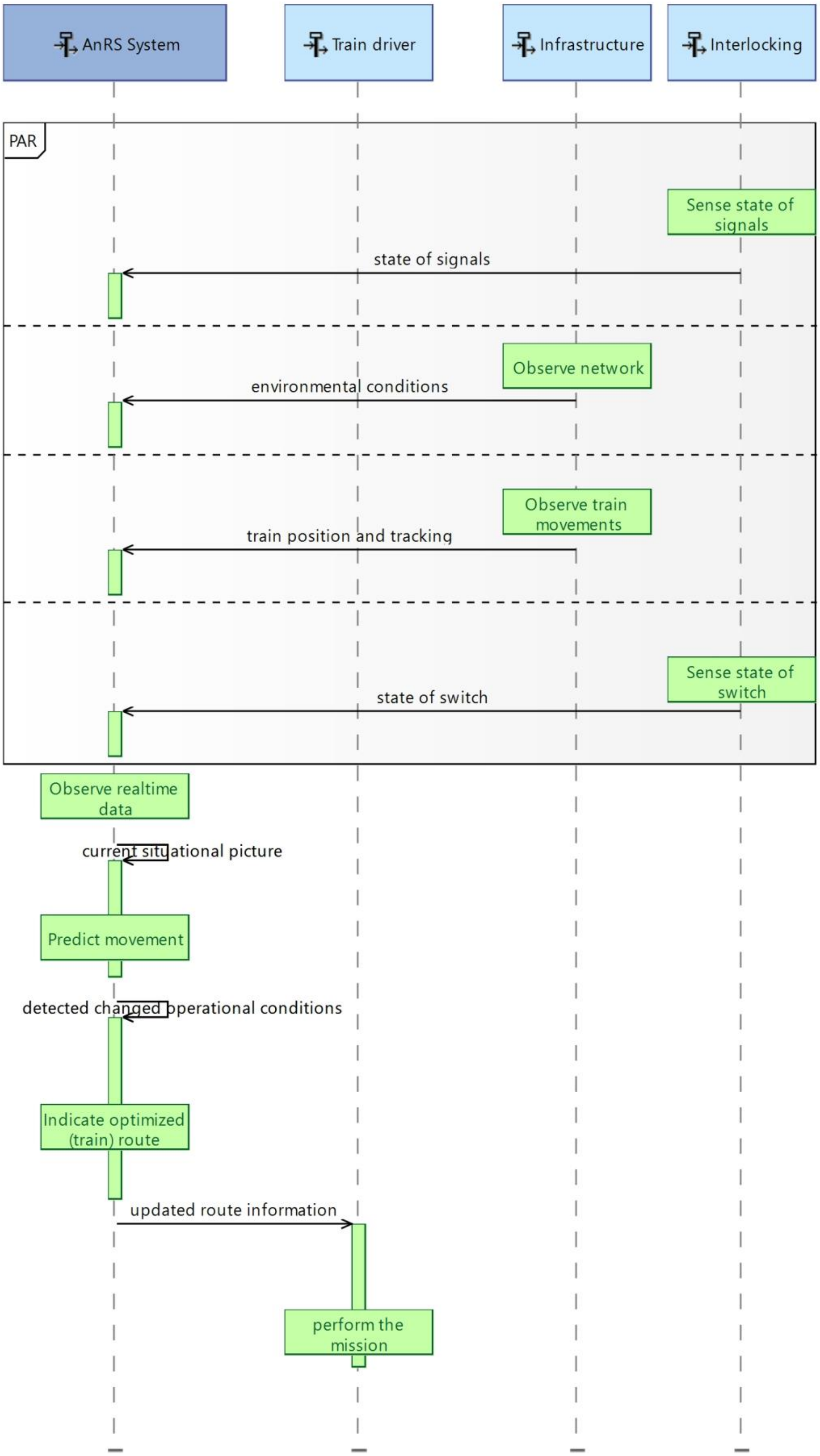


**Figure 33: Exchange Scenario Diagram**

Owned messages:

- state of signals
- environmental conditions
- train position and tracking
- state of switch
- current situational picture
- detected changed operational conditions
- updated route information

#### 4.4.6.5. Scenario: [ES] Conflict resolution and coordination

Description:

*No description*

Allocated capability:

- Conflict resolution and coordination

Representation:

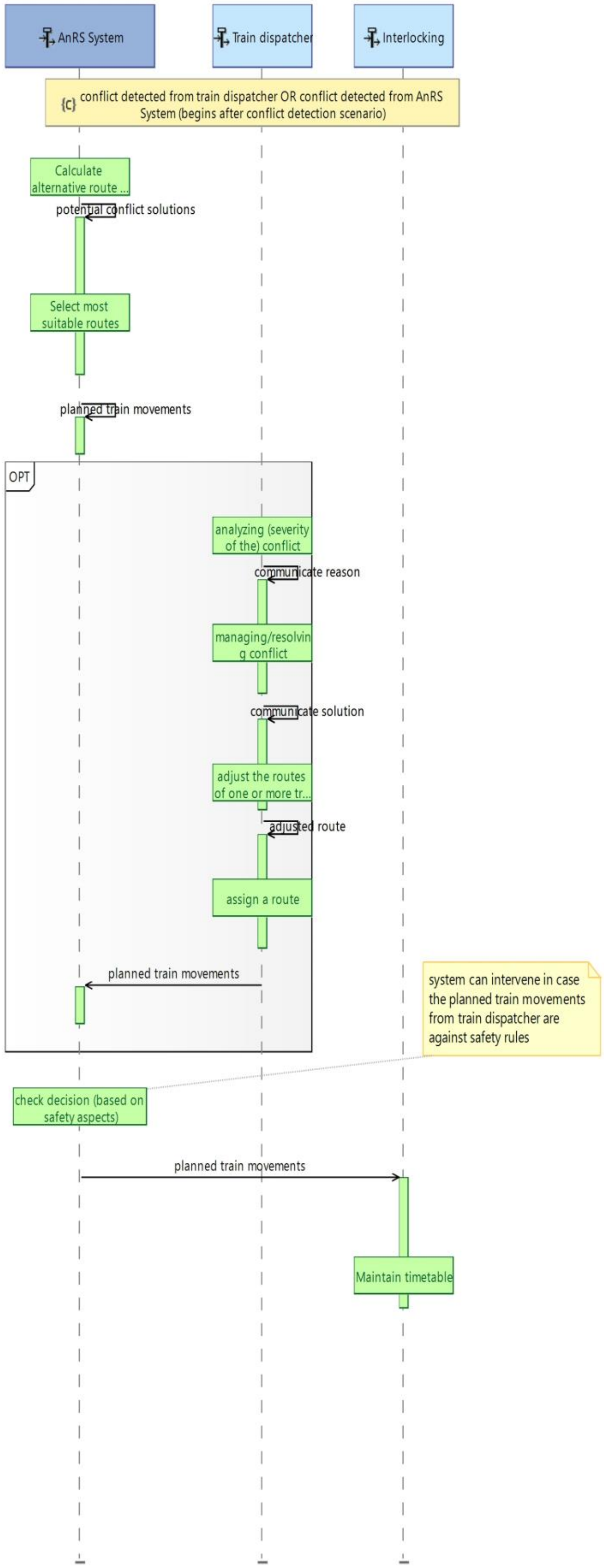


**Figure 34: Exchange Scenario Diagram**

Owned messages:

- potential conflict solutions
- planned train movements
- communicate reason
- communicate solution
- adjusted route
- planned train movements
- planned train movements

## 4.5. SYSTEM ARCHITECTURE AND CONCEPTUAL INTEGRATION

The Figure 35shows a possible high-level integration of the AnRS into the existing traffic management infrastructure. The aim is to centralize the decision-making processes and to place the AnRS as middleware between traffic management and interlocking. The challenge is to define an intelligent and autonomous system that can be integrated into the existing infrastructure (retrofitting) by using the existing interfaces (SCI-CC) and continue to operate them unchanged. In doing so, we realize that two additional components are also required. An environment capturing unit and an information management system that ensures the correctness of the underlying database.

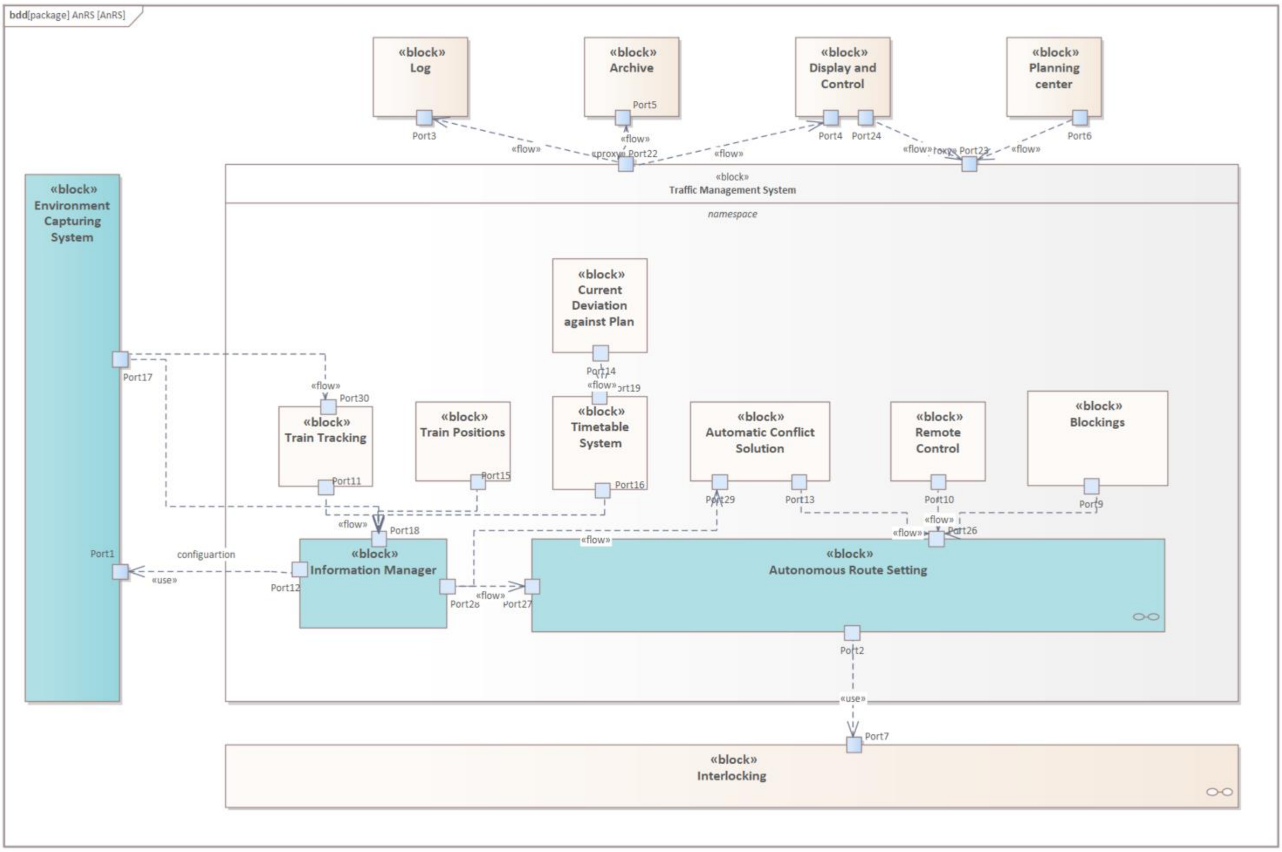


Figure 35: Overview of Traffic Management Architecture with AnRS Integration

Open issues that need to be further specified in the concept and then realized at a logical and functional level are the topics of how distributed AnRS instances communicate with each other and coordinate decisions. We are currently assuming a distributed, decentralized system that enables greater flexibility and scalability. For the AnRS to work properly to support components are relevant an Environment Capturing System to detect objects on the track or if a track is free from other trains in general. And an information management component.

**Environment Capturing System**

The Environment Capturing System is relevant for external data. The Environment Capturing System observe the trackside and refers to all information that is used to monitor and digitally describe the environmental conditions that are relevant for the operational level. In addition to sensor data, this could also include semistatic data such as the measurement of the environment as an HD map. Semi-static because the exact reference points and positions of existing data must be stored precisely for an HD map. Due to weather influences, for example, certain objects may move slightly, and a mechanism must therefore be developed to update this supposedly static information. Weather influences also play a decisive role in the validity and reliability of sensor data. LiDAR, for example, performs worse in poor weather conditions than in good weather. This must also be considered when monitoring and safeguarding automatic processes and the weather information must be processed accordingly. To ensure that the train operates correctly assesses critical situations, the situation images between the train and the infrastructure must be compared, which requires the train data to be transmitted to the trackside. The topic of interface and processing also plays an important role here.

**Information Manager**

The information manager guarantees that the same information processing chain, i.e. pre-processing, fusion, etc., provides the same information for all subcomponents so that the basis for decision-making has no effect on misconduct. Accordingly, the subcomponents can log in to the information manager and request the data required for their decision-making, whereby the information manager ensures a stream with the corresponding data.

### 4.5.1. Autonomous Route Setting

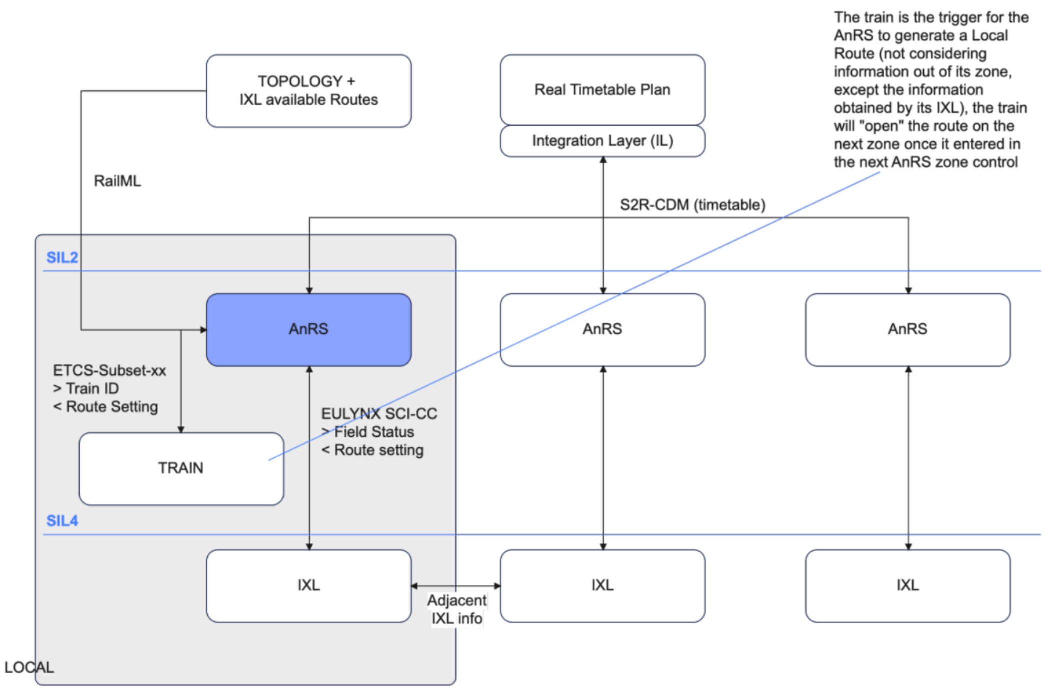


**Figure 36: Microscopic AnRS Architecture (local)**

In a world where efficient transportation is paramount, the concept of Automatic Route Setting (ARS) takes center stage. However, in certain scenarios where centralized control systems like Train Control (CTC) or Traffic Management Systems (TMS) are absent, innovative solutions are required. This is where the notion of a Real Timetable Plan (RTTP) and its integration with AnRS systems becomes essential.

The fundamental premise of this approach lies in its ability to manage train routes within a localized area of control, all without the presence of a centralized control entity like CTC or TMS. Instead, it relies on the deployment of AnRS mechanisms, functioning on a smaller scale and employing localized communication protocols such as T2G/G2T.

At the heart of this system is the RTTP, a comprehensive plan that serves as a blueprint for train operations within the designated area. Unlike traditional centralized control systems, which dictate routes from a central command center, the RTTP allows for decentralized decision-making, empowering local entities to manage routing based on real-time data and predefined schedules.

The integration of AnRS with RTTP brings forth a sophisticated routing mechanism. When a train enters the system, it communicates its identification number, enabling the AnRS to match it with the RTTP and access a library of predefined routes specific to the area managed by a single IXL. This process ensures that each train is directed along the most efficient path to execute the plan outlined in the RTTP.

Furthermore, the AnRS system facilitates collaboration among trains within the network. Through decentralized communication protocols, trains exchange information, allowing for the optimization of routes based on real-time conditions and dynamic adjustments to the RTTP. This collaborative approach not only enhances efficiency but also improves resilience, as trains can adapt to unforeseen circumstances such as delays or disruptions.

The utilization of AnRS with RTTP offers several key benefits in various operational scenarios. One such scenario is the management of shunting, depot, and station operations. In environments where centralized control systems are absent, trains are automatically routed to their designated tracks or platforms based on train detection. This automation streamlines operations, minimizing manual intervention and reducing the risk of errors.

Another critical application of this approach is in decentralized traffic management. In situations where there is no centralized control system along the line, each AnRS unit communicates directly with others at the edge, collaboratively establishing routes based on predefined RTTP guidelines. This decentralized approach fosters agility and adaptability, allowing trains to navigate complex network configurations with ease.

The integration of AnRS with RTTP also opens up opportunities for future advancements in rail transportation. By leveraging technologies such as artificial intelligence and machine learning, the system can continuously optimize routes, considering factors such as traffic patterns, weather conditions, and infrastructure capacity. This predictive capability not only improves efficiency but also reduces risks by proactively mitigating risks.

Furthermore, the data generated by the AnRS-RTTP system provides valuable insights into train operations and network performance. By analyzing this data, rail operators can identify trends, optimize resource allocation, and make informed decisions to improve overall system reliability and passenger satisfaction.

In conclusion, the integration of Autonomous Route Setting with Real Timetable Planning represents a paradigm shift in rail transportation. By decentralizing control and leveraging collaborative communication protocols, this approach offers a scalable and adaptable solution for managing train routes in diverse operational environments. As technology continues to evolve, so too will the capabilities of AnRS-RTTP systems, driving innovation and shaping the future of rail transportation.

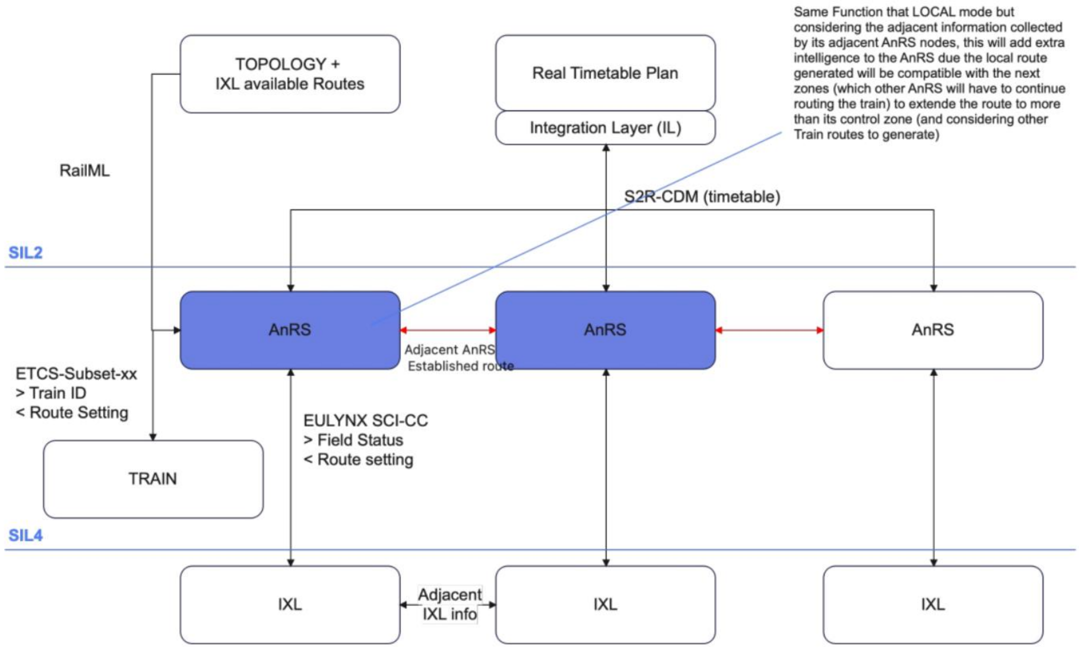


**Figure 37: Macroscopic AnRS Architecture (global)**

# 5. OPERATIONAL CONCEPT ANALYSIS

For the evaluation of the concept, an evaluation is to take place in two steps. First, the use cases derived in the previous chapters will be applied, and the function of the AnRS will then be explained through validation, using the example of the accident reports discussed earlier. Finally, a prototype of a possible AnRS will be integrated into the existing INDRA lab environment to demonstrate the compatibility of an AnRS with existing infrastructure.

## 5.1. SCENARIO AND DEMONSTRATION PLANNING (CASE STUDY)

### 5.1.1. Step 1 – Theoretical Use Cases Evaluation

**Use Case 1 – Passing loop**

In a decentralized system, trains and trackside equipment (such as switches and signals) communicate directly with each other to determine the best routes through a section of track, fulfilling a precharged operational plan. The system would operate on the basis of predefined rules and algorithms that allow each component to make decisions based on real-time conditions.

In the first use case we assume a single-track railway line with several passing loops (short stretches of double track where trains can pass each other). Trains traveling in opposite directions need to use these loops to avoid collisions. With a decentralized AnRS system we could handle this situation as follow, each train is already equipped with a communication system that broadcast its position, speed, and direction to nearby trains and trackside equipment. On the trackside we have sensors detect the presence of trains approaching a passing loop. As two trains approach a passing loop from opposite directions, the loop's local AnRS system (which control switches and signals via the interlocking) receives information from both trains. The system evaluates factors like the speed and distance of each train, as well as the occupancy of the loop. Different situations can occur:

1. If one train is significantly closer to the loop, the system might decide to allow that train to enter the loop first, while signaling the other train to slow down or stop at a signal. The trains themselves could also be involved in the decision-making process. In that case, Train A might broadcast a signal indicating that it is close to the loop and moving quickly. Train B, receiving this signal, might automatically adjust its speed to ensure it arrives at the loop after Train A has passed.
2. both trains are equally close, the system could apply a rule, such as prioritizing the train with the higher priority (e.g., a passenger train over a freight train).

Once the decision is made, the local AnRS system automatically sets the switches to route the first train into the loop and the second train onto the main track. After Train A passes, the system resets the switches to allow Train B to continue on the main line.

In case of communication failure or other issues, the system could revert to fail-safe modes. For example, it might default to rerouting trains by coordinate with other AnRS systems until the situation is.

Consider a single-track route with a passing loop. Train A and Train B are heading in the same direction. Both are about to approach a passing loop. Train A and Train B broadcast their positions. The system at the passing loop detects that Train A comes closer. The loop's AnRS system decides to prioritize Train A because of higher maximum speed. It sets the switches to route Train A onto the main line through the loop, while Train B is signaled to change to the passing loop. Train B receives the slow down signal and begins braking. Meanwhile, Train A passes through the loop and continues the journey. After Train A clears the loop, the system resets the switches and signals Train B to proceed as well.

The system can quickly adapt to real-time conditions, such as delays or unexpected stops. Adding new loops or tracks can be done without significant changes to a central system, as each section operates semi-independently.

But ensuring smooth communication between all parts of the system is crucial, and any breakdown could lead to delays or accidents. Rigorous testing and fail-safes are needed to ensure that the decentralized system is as safe as a traditional centralized one. In summary, a decentralized route setting system for passing tracks could allow trains and trackside equipment to make real-time decisions about routing, improving flexibility and efficiency, particularly in complex or variable environments like single-track railways with multiple passing loops.

**Use Case 2 – Rail Junctions**

At a rail junction, multiple tracks converge, and trains need to cross, merge, or diverge safely and efficiently. In a decentralized system, trains and junction control equipment would communicate directly to negotiate safe passage through the junction, without relying on a central control center.

In the second use case we assume a rail junction where four tracks intersect, forming a cross. Trains on any of the tracks can potentially cross paths, merge onto another track, or diverge. A decentralized AnRS system might handle this as follow: each train approaching the junction is equipped as in use case 1. The trackside is equipped as in use case 1 as well. The junction has a local AnRS system equipped with a trained logic for managing train movements based on real-time environmental and train data. When multiple trains approach the junction, the local AnRS system evaluates which trains have priority based on factors like their speed, distance from the junction, and their intended routes.

Once the system determines the optimal sequence for train movements, it sets the switches accordingly and updates signals to reflect the allowed routes. For example, if Train A is cleared to cross straight through the junction, the switches are set to allow this path, and signals for Train B are set to red. After Train A clears the junction, the system resets the switches and signals to allow Train B to proceed. In case two trains are approaching the junction from conflicting directions at nearly the same time, the system might instruct one train to take a different route if possible (e.g., diverting to a parallel track) or to wait at a signal until the other train has passed.

Consider a busy urban rail junction where several commuter and freight trains intersect. At peak hours, trains from four different lines (North, South, East, and West) approach the junction, some intending to cross straight through, while others need to turn onto different track, the AnRS system could value priorities based on the trained model and perform the conflict resolution from a more reactive perspective.

In a more complex scenario, imagine a major rail interchange where several tracks cross at different levels (e.g., some tracks go over or under others). Here, the decentralized system could manage vertical and horizontal movements, ensuring that trains can safely cross or merge at different levels.

In summary, a decentralized route setting system for rail junctions would allow trains and local control systems to work together to manage complex movements through the junction. This negotiation could based on complex cost functions (local and global ones) to decide which decision has the lowest impact on the overall cost function.

### 5.1.2. Step 2 – Lab Environment Demonstration

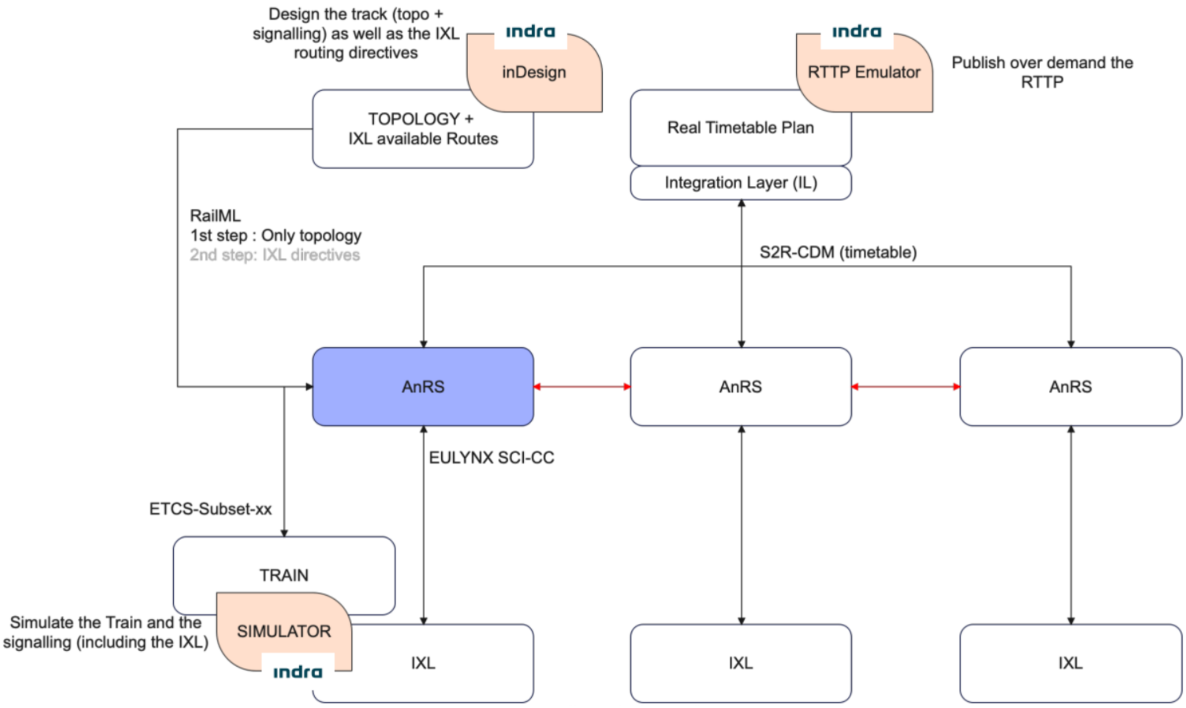


**Figure 38: Planned Test setup for AnRS Concept - LAB Environment**

In Figure 38 the planned demonstration and evaluation setup is designed. To show that the AnRS as a decentralized system could work, we plan to embed the system into a realistic simulation environment and connect it to the existing interfaces and show he approach based on the described use cases and scenarios. For the evaluation we have defined the following requirements to the simulation especially to perform verification and validation tasks to ensure the simulation accurately reflects real-world conditions and rigorously tests the system. The aim is to model and re-enact the accident scenario in order to use it as state-of-the-art technology and demonstrate the advantages of AnRS in these special cases.

On the one hand the simulation must have high fidelity, meaning it should closely replicate the actual environment and components involved in the system being tested. This includes accurate modelling of physical elements like tracks, signals, and trains, as well as logical elements such as interlockings and control systems. The simulation must also be deterministic, ensuring that given the same inputs, it will always produce the same outputs. This is essential for verifying that the system behaves consistently under the same conditions.

On the other hand, the simulation should also support exhaustive testing, allowing for a wide range of scenarios, including edge cases and unlikely events that might not be easily replicated in the real world. It should be capable of simulating both normal operations and failure modes, including rare or critical failures, to ensure the system can handle any possible situation safely.

The simulation's ability to integrate real-world data and systems helps to compare the results with existing situations like the accidents that we analysed during this concept study. This allows the validation process to account for real-life variability and helps verify that the system will function correctly when deployed in the actual environment. The simulation should also be able to produce detailed logs and reports of all test scenarios and outcomes, which are necessary for thorough analysis and documentation during the verification and validation process.

# 6. CONCLUSION

Due to the rapid technological advancements in digitalization over the past few years, it has become increasingly feasible to revisit and implement earlier approaches that were once impractical due to insufficient or unfavourable conditions. One such approach is the Autonomous Route Setting (AnRS) System, a central focus of this concept study. This study, as outlined in the introduction, delves into the development of advanced systems for railway operations, with a specific emphasis on achieving GoA4. GoA4 represents the highest level of automation, where human intervention is minimized, and the system operates with full autonomy. This ambitious scope underscores the potential of AnRS in optimizing railway operations, aligning perfectly with the broader trends in automation highlighted in the introduction.

This study begins with an exploration of GoA4, which serves as the foundation for understanding the critical role of the AnRS. The introduction rightly emphasizes the importance of Automatic Route Setting (ARS) in optimizing railway operations. The AnRS builds upon this by representing a decentralized edge solution that can make localized decisions for individual components, such as railway switches, and collaborate with other systems to find optimal solutions. This capability is crucial for achieving the high levels of automation discussed in the introduction.

A significant advantage of the AnRS is its compatibility with existing infrastructure. By adhering to established interfaces and standards, the system can be seamlessly integrated into current setups, making it ideal for retrofitting projects—a point also emphasized in the study. The integration of AI, as mentioned in the introduction, plays a crucial role in automize the decision-making within the AnRS framework. These technologies enable the system to process real-time observation data, make adaptive decisions, and contribute to the overall efficiency and safety of railway operations.

The introduction's discussion on global expansion and the influence of international trends is highly relevant to the AnRS, as the system is designed to be adaptable across various geographic and regulatory environments. This adaptability is crucial for supporting the expansion of global railway networks and ensuring that the system can be effectively deployed in diverse settings. The emphasis on considering cybersecurity aspects is also well-reflected in the AnRS design. Cybersecurity is a top priority, given the complex and critical nature of automated railway systems. The study addresses potential threats and outlines measures to protect the system from cyber risks, ensuring the integrity and reliability of the AnRS.

As highlighted in the "State of the Art" section of the study, the examination of GoA4's functional requirements and current implementations provides a critical foundation for understanding how these technologies are applied in practice. The study also addresses regulatory considerations essential for compliance, which are vital for ensuring that the AnRS can operate within existing legal frameworks. The analysis of ARS functionality, including components such as the Smart Wayside Object Controller and Radio Driving Mode, further contextualizes how the AnRS fits into the broader landscape of railway automation.

Moreover, the system architecture and conceptual integration are presented in a way that aligns with existing infrastructure and interlocking interfaces, ensuring seamless integration into current operations without disruptions. This aspect of the study is crucial for practical implementation, as it demonstrates that the AnRS can be deployed in real-world settings without requiring extensive modifications to existing systems.

An operational concept analysis, based on use cases and a demonstration case study, is included in the study to illustrate the practical application of the AnRS system. This case study provides a concrete example of how the proposed system operates in a real-world setting, bridging the gap between theoretical concepts and their practical application. In the last year of FP2-R2DATO the focus is on simulative verification and validation. By employing advanced simulation techniques, the system's performance will be tested under various scenarios to ensure its reliability and effectiveness before any real-world deployment. This step is particularly crucial in the railway domain, where safety and precision are paramount.

As the work package for the AnRS is still ongoing, the upcoming activities planned for the remaining project year, will serve as preparation for Wave 2. The next steps include the development of a prototype implementation of the AnRS, which will be integrated into INDRA's laboratory environment. This prototype will undergo further testing and refinement, paving the way for eventual deployment in real-world settings.

# 7. REFERENCES

[1] J. Weimer, C. Ulrich, M. Conzelmann, T. Fleck, M. R. Zofka, and M. Grünhäuser, “Managed automated driving: a new way for safe and economic automation,” in *27th ITS World Congress*, 2021. [Online]. Available: https://elib.dlr.de/144893/

[2] A. Correa *et al.,* “TransAID Deliverable 5.2: V2X-based cooperative sensing and driving in Transition Areas (second iteration),” 2020. [Online]. Available: https://elib.dlr.de/140794/

[3] E. Israel and G. Cohen-Blankshtain, “Testing the decentralization effects of rail systems: Empirical findings from Israel,” *Transportation Research Part A: Policy and Practice*, vol. 44, no. 7, pp. 523–536, 2010, doi: 10.1016/j.tra.2010.03.021.

[4] M. Kii, V. Vichiensan, C. Llorca, A. Moreno, R. Moeckel, and Y. Hayashi, “Impact of Decentralization and Rail Network Extension on Future Traffic in the Bangkok Metropolitan Region,” *Sustainability*, vol. 13, no. 23, p. 13196, 2021, doi: 10.3390/su132313196.

[5] E. Marcelli and P. Pellegrini, “Literature Review Toward Decentralized Railway Traffic Management,” *IEEE Intell. Transport. Syst. Mag.*, vol. 13, no. 3, pp. 234–252, 2021, doi: 10.1109/MITS.2020.2970180.

[6] J. Rasmussen, “Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models,” *IEEE Trans. Syst., Man, Cybern.*, SMC-13, no. 3, pp. 257–266, 1983, doi: 10.1109/TSMC.1983.6313160.

[7] L. Fendrich and W. Fengler, *Handbuch Eisenbahninfrastruktur*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2019.

[8] P. Antsaklis, “Autonomy and metrics of autonomy,” *Annual Reviews in Control*, vol. 49, pp. 15–26, 2020, doi: 10.1016/j.arcontrol.2020.05.001.

[9] M. Simmler and R. Frischknecht, “A taxonomy of human–machine collaboration: capturing automation and technical autonomy,” *AI & Soc*, vol. 36, no. 1, pp. 239–250, 2021, doi: 10.1007/s00146-020-01004-z.

[10] N. Jeffrey, Q. Tan, and J. R. Villar, “A Review of Anomaly Detection Strategies to Detect Threats to Cyber-Physical Systems,” *Electronics*, vol. 12, no. 15, p. 3283, 2023, doi: 10.3390/electronics12153283.

[11] J. Karlin, S. Forrest, and J. Rexford, “Autonomous security for autonomous systems,” *Computer Networks*, vol. 52, no. 15, pp. 2908–2923, 2008, doi: 10.1016/j.comnet.2008.06.012.

[12] A. P. Williams and P. D. Scharre, Eds., *Autonomous systems: Íssues for defence policymakers*. Norfolk, Va.: HQ Sact, 2016.

[13] S. Miller and M. Baertsoen, *Alstom demonstrates fully autonomous driving of a shunting locomotive in the Netherlands*, 2022. [Online]. Available: https://www.alstom.com/press-releases-news/2022/11/alstom-demonstrates-fully-autonomous-driving-shunting-locomotive-netherlands

[14] R. L. French and G. M. Lang, “Automatic route control system,” *IEEE Trans. Veh. Technol.*, vol. 22, no. 2, pp. 36–41, 1973, doi: 10.1109/T-VT.1973.23525.

[15] H. Teshima, “Railway track layout modelling and its application to an automatic route setting system,” in *Computers in Railways XIV*, Rome, Italy, 2014, pp. 75–86.

[16] M. Kuhn, “Automatic route setting integrated in a modern traffic management system,” in *International Conference on Developments in Mass Transit Systems*, London, UK, 1998, pp. 140–145.

[17] S. Iwnicki, “Future trends in railway engineering,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 223, no. 12, pp. 2743–2750, 2009, doi: 10.1243/09544062JMES1545.

[18] A. Ucar, M. Karakose, and N. Kırımça, “Artificial Intelligence for Predictive Maintenance Applications: Key Components, Trustworthiness, and Future Trends,” *Applied Sciences*, vol. 14, no. 2, p. 898, 2024, doi: 10.3390/app14020898.
[19] P. Singh, Z. Elmi, V. Krishna Meriga, J. Pasha, and M. A. Dulebenets, “Internet of Things for sustainable railway transportation: Past, present, and future,” *Cleaner Logistics and Supply Chain*, vol. 4, p. 100065, 2022, doi: 10.1016/j.clscn.2022.100065.
[20] V. J. Hodge, S. O'Keefe, M. Weeks, and A. Moulds, “Wireless Sensor Networks for Condition Monitoring in the Railway Industry: A Survey,” *IEEE Trans. Intell. Transport. Syst.*, vol. 16, no. 3, pp. 1088–1106, 2015, doi: 10.1109/TITS.2014.2366512.
[21] A.-Q. Gbadamosi *et al.,* “IoT for predictive assets monitoring and maintenance: An implementation strategy for the UK rail industry,” *Automation in Construction*, vol. 122, p. 103486, 2021, doi: 10.1016/j.autcon.2020.103486.
[22] P. Fraga-Lamas, T. M. Fernández-Caramés, and L. Castedo, “Towards the Internet of Smart Trains: A Review on Industrial IoT-Connected Railways,” *Sensors (Basel, Switzerland)*, vol. 17, no. 6, 2017, doi: 10.3390/s17061457.
[23] C. A. Brebbia *et al., Eds., Computers in Railways XIV*: WIT PressSouthampton, UK, 2014.
[24] J. Otegui, A. Bahillo, I. Lopetegi, and L. E. Diez, “A Survey of Train Positioning Solutions,” *IEEE Sensors J.*, vol. 17, no. 20, pp. 6788–6797, 2017, doi: 10.1109/JSEN.2017.2747137.
[25] E. Bernal, M. Spiryagin, and C. Cole, “Onboard Condition Monitoring Sensors, Systems and Techniques for Freight Railway Vehicles: A Review,” *IEEE Sensors J.*, vol. 19, no. 1, pp. 4–24, 2019, doi: 10.1109/JSEN.2018.2875160.
[26] D. Ristić-Durrant, M. Franke, and K. Michels, “A Review of Vision-Based On-Board Obstacle Detection and Distance Estimation in Railways,” *Sensors (Basel, Switzerland)*, vol. 21, no. 10, 2021, doi: 10.3390/s21103452.
[27] P. Dixit, P. Bhattacharya, S. Tanwar, and R. Gupta, “Anomaly detection in autonomous electric vehicles using AI techniques: A comprehensive survey,” *Expert Systems*, vol. 39, no. 5, 2022, doi: 10.1111/exsy.12754.
[28] S. Khayyam, F. Ponci, J. Goikoetxea, V. Recagno, V. Bagliano, and A. Monti, “Railway Energy Management System: Centralized–Decentralized Automation Architecture,” *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 1164–1175, 2016, doi: 10.1109/TSG.2015.2421644.
[29] A. Fraszczyk and M. Marinov, Eds., *Sustainable Rail Transport*. Cham: Springer International Publishing, 2019.
[30] F. C. Nemtanu and M. Marinov, “Digital Railway: Trends and Innovative Approaches,” in *Lecture Notes in Mobility, Sustainable Rail Transport*, A. Fraszczyk and M. Marinov, Eds., Cham: Springer International Publishing, 2019, pp. 257–268.
[31] M. James, L. Rodrigue, and A. El-Geneidy, “Toward a Better Understanding of the Construction Impacts of a Light Rail System in Montréal, Canada,” *Transportation Research Record: Journal of the Transportation Research Board*, 2024, doi: 10.1177/03611981241242362.
[32] R. Dipl.-Ing. Wolf and H.-G. Dr.-Ing. Langer, “GoA4-Readiness – Herausforderungen für zukünftige Fahrzeuggenerationen,” *ZEVRail*, 1-2, no. 146, pp. 4–9, 2022.
[33] C. Salander, *Das Europäische Bahnsystem*. Wiesbaden: Springer Fachmedien Wiesbaden, 2019.
[34] E. Schnieder and L. Schnieder, *Verkehrssicherheit*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.
[35] F. Hagemeyer, M. Preuß, M. Meyer zu Hörste, C. Meirich, and L. Flamm, *Automatisiertes Fahren auf der Schiene*. Wiesbaden: Springer Fachmedien Wiesbaden, 2021.

[36] L. Jerónimo Vázquez, F. Parrilla Ayuso, S. Jiménez Gómez, J. Mattisson, B. Losada, and M. Alba, *Specifications and Test Strategy: Deliverable D11.1*.
[37] D. López *et al., Review and Conclusions of Demonstrators at TRL4: Deliverable D11.3*.
[38] P. Philippe, *Review and Conclusions of Demonstrators at TRL6: Deliverable D12.2*.
[39] J. Pachl, *Systemtechnik des Schienenverkehrs: Bahnbetrieb planen, steuern und sichern ; mit 13 Tabellen und Beispielen,* 4th ed. Stuttgart, Leipzig, Wiesbaden: Teubner, 2004.