ERJU SYSTEM PILLAR

T3-Interface Specification TMSCCS

T3-Interface Specification TMSCCS

Author(s)	NANNI Marco , Krista Matthias (I-NAT-GST-CCS-EXT - Extern) , Uhlich Sebastian (I-NAT-GST-CCS-EXT - Extern)
Abstract	Describe the context of application and the design principles of the interface between TMS and CCS, its general structure and the data model
Config Item	System Interface Description
Document ID	30 Deliverables/T3-Interface Specification_TMS_CCS#717104 T3-Interface Specification TMSCCS
Classification	Public
Status	In Review by System Pillar
Version	1.6
Revision	717104
Last Change Date	29.09.2025
Copyright	Brussels: Europe's Rail Joint Undertaking, 2025

This work is currently a work in progress. The content presented is subject to change as it undergoes further review, refinement, and development. Please do not consider this version as final or authoritative.

INFO: History table is not displayed, because this document is in status **doc_contentApproval**.
RULE: History table is not displayed, in statuses: { draft doc_open doc_inprogress doc_contentApproval doc_contentDecision }
CONTACT: For more information contact Administrator

Review Description

Approvals	NANNI Marco : Waiting , Simone Brezzi : Waiting , Knoedl Herbert (INFRA.Netzzugang) : Waiting , Ghersi Mirko : Waiting , Patrick Konix : Waiting , CANEPA RENZO : Waiting , BO MARCO : Waiting , Bence Jenőfi : Waiting , POURCHIER Jean Marc (SNCF RESEAU / SIEGE SNCF RESEAU / DIR EUROPE) : Waiting
Type of Approval	Document Review

Approval Description

Approvals	Knoedl Herbert (INFRA.Netzzugang) : Waiting
Type of Approval	Document Approval

Table of Contents

Table of Figures

1 Preamble

1.1 Purpose

The main goal of this document is to show the complete set of data to be exchanged and how they are organized in structures and messages, keeping full alignment with the data model specified by the Task 2 transversal team. Some examples are provided as a guideline, to show how data can be represented. **[**SPT3TMS-16768 **]**

1.2 Intended audience

This document is intended for all stakeholders involved in the development, implementation, and operation of TMS and CCS systems (e.g. Business stakeholders, End users, Development and engineering teams, Assessors, etc.) **[**SPT3TMS-16770 **]**

1.3 Document Context

This document aims to approach the interface specification between the TMS and Traffic CS subsystems, from a functional viewpoint.
A general description of the context of application and a short description of the functions allocated to the peer systems which are connected is also given. **[**SPT3TMS-16769 **]**

1.4 Glossary

1.4.1 Terms and definitions

N.A.

1.4.2 Abbreviations

Term	Definition
AoC	Area of Control
AoE	ATO over ETCS
APS-SM	Advanced Protection System Safety Manager
ARS	Automatic Route Setting
ATO	Automatic Train Operation
ATO-OB	ATO-OnBoard
ATO-TS	ATO-Trackside
CCS	Control Command and Signalling
CIA	Confidentiality, Integrity, Availability
CMS	Capacity Management System

Term	Definition
CTC	Centrale Traffic Control
DPS	Drive Protection Sections
EAL	Execution and Adaptation Layer
ETCS	European Train Control System
FFFIT	Form-Fit Functional Interface Specification
FIT	Functional Interface Specification
GoA	Grade of Automation
IPM-ERM	Evaluation Manager
MP	Movement Permission
NFRs	Non-Functional Requirements
ORA	Operational Restriction Area
OWA	Operational Warning Area
PES	Plan Execution System
RAMSS	Reliability, Availability, Maintainability, Safety and Security
SAR	Safety-relevant application rules according to EN 50716
TCR	Train Capability Report
TMS	Traffic Management System
TPS	Trackside Protection System
TS	TrackSide

[SPT3TMS-16861 **]**

2 History Of Changes

Nr.:	Changes:	Leaders/ Authors:
1.1	Initial Draft - SCI-OP Concepts	Christoph Growe
	Comments	Martin Kemkemer,

Nr.:	Changes:	Leaders/ Authors:
		Ulrich Schöni
1.2	Second Draft - Diagrams Update	Raghda Mohamed
	Comments	Martin Kemkemer, Ulrich Schöni
1.2.1	Third Draft - Upstream & Downstream Update. Aligned and integrated SCI-OP data model with TCCS SD1 for the following messages. • Operational Plan Execution Request (only: Operational Plan Movement) • Operational Plan Execution Response • Operational Plan Execution Report • Train Unit Data Report, Train Unit State Report, Train Unit Position Report • Track Allocation Report, Track Allocation Removal • Field Element State Report	Martin Kemkemer, Sebastian Uhlich
1.3	Fourth Draft - Upstream & Downstream Update. Aligned and integrated SCI-OP data model with TCCS SD1 for the following messages: • Event Timings • Operational Plan Execution Request • Train Unit Data Report, Train Unit State Report, Train Unit Position Report • Operational Plan Execution Response • Track Occupation Report • DPS Group State • Operational Plan Warning Measure	Martin Kemkemer, Sebastian Uhlich
1.3.1	Minor fixes and improvements based on approval review comments.	Martin Kemkemer, Sebastian Uhlich
1.3.2	• Fixes to Mirror-Group Review • Minor improvements to chapter structure • Adding several new attributes (scheduledTrainLength in TrainUnit, excludedDPSGroupsForFlankProtection) • Added new Chapter "Support two different configurations of TMS <> CSS"	Martin Kemkemer, Sebastian Uhlich

Nr.:	Changes:	Leaders/ Authors:
	• Updated Chapter 3.2.8 Support of Publish-Subscribe-Principle	
1.4	• Introduced Event Links • Added Chapter 5 • Renamed "Train Occupation State" by "Track Occupation State" in some diagrams • Added new Design Principle for "Data Sychronization" • Answered comments raised in Polarion with consequent integrations • Modified wording and adaptation to Traffic CS new architecture. Changed pictures at chapters 3.1 and 3.4 • Removed RCA references from the architecture • Added two additional parameters, atoInhibition and Max Current to the Restrictions Aspects, according to what agreed with Traffic CS and SD1 teams for Data Model • Aligned data structures with Data model namings changes according to ERA Ontology	Marco Nanni
1.5	Document aligned with the defined criteria in CMP.	Marco Nanni
2.0	• Added improvements according to the comments received by Mirror Group as part of the 2024-2025 review • Integration of Data Model updates to align with TAF-TAP TSI as suggested by Era and performed by TCCS team • Clarified some parts of the document to better describe some relevant concepts • Described the purpose of the proposed messages • Added "Table of Content" and "Table of Figures" as requested by CMP standards (this implies that existing document titles shifted of 2 indexes") • Alignment to the new release of TCCS data model at sept 2025. Major changes are: ◦ revise Train Object class ◦ removed DPS classes and replaced with SwitchableTracksideAsset related classes. Restructured TrainObject class. ◦ replaced Track Occupation state class (but the message name remains the same) with UnresolvedTrackboundObject class ◦ Revised MP class	Marco Nanni

3 Introduction

This document describes the Communication Interface between TMS and CCS/Traffic CS, “TMS <> CCS” (referred to as “SCI-OP” in RCA documents). It is a further development of the very well worked out document Concept SCI-OP RCA (document id: RCA.Doc.31) by RCA Group and largely follows that description. This further development aims to align and make consistent the interface specification with the overall architecture and data model elaborated in the context of system Pillars initiative. **[**SPT3TMS-9723 **]**

Documents for a deeper understanding of TMS <> CCS are available on RCA website (Link: RCA documentation). The current version of this document may not yet contain all chapters required. The focus of this document lays on the concept for an interface between TMS and CCS. Further chapters can be added by the alignment with the steering board. **[**SPT3TMS-9722 **]**

The present document draws heavily on prior work performed in this area by both individual railway companies and within the RCA. For this reason, certain terminological and conceptual inconsistencies may occur between the concepts described herein and other documents (e.g., System Concept TMS) previously presented. It is the authors’ assumption that these will be reconciled as work on TMS definition progresses and details are further refined. **[**SPT3TMS-16663 **]**

This document represents a Functional Interface Specification (FIT), as it focuses on the application layer only. It doesn't approach all details required by a FFFIT, specifically the communication protocol, to keep disjointed the specification of **what** shall be done from **how** it shall be achieved. Nevertheless, it shall be necessary to plan this step later, in agreement with other interested stakeholders, as Traffic CS domain. **[**SPT3TMS-16664 **]**

4 Standard Communication Interface TMS <> CCS

This chapter describes the key characteristics of the Interface between TMS and CCS its design principles, and addresses migration.

4.1 Introduction

The interface between TMS and CCS, referred to as “Standard Communication Interface Operational Plan (SCI-OP)” in the Reference CCS Architecture (RCA) is located at the system boundary of CCS and connects TMS with the CCS’ ATO Trackside (ATO-TS) and Plan Execution System (PES), (previously referred to as Execution and Adaptation Layer, EAL) capabilities. As such, it serves to ensure the flow of data between systems while encapsulating their internal logic. **[**SPT3TMS-9717 **]**

Figure 1, below, depicts the TMS <> CCS interface’s role between the systems for dispatching and execution.

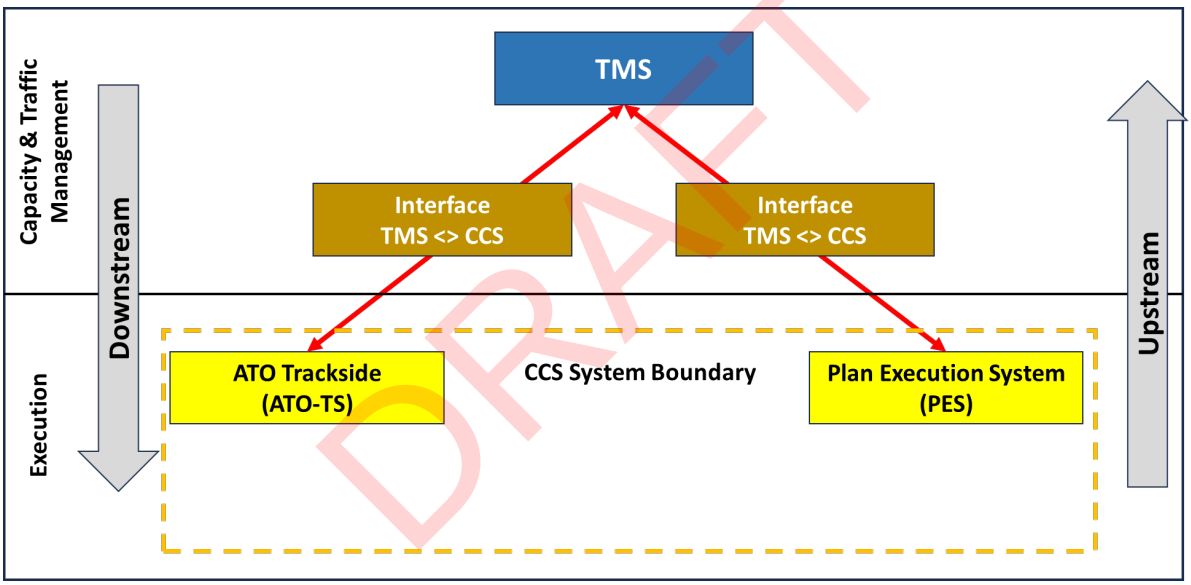


Figure 1 Scope of CCS interface TMS <> CCS

Describing TCS architecture is not a goal of this document, and the drawing aims only to show which are the peers subsystems which TMS connects to. **[**SPT3TMS-16096 **]**

ATO Trackside System (ATO-TS) re-elaborates the Operational Plan received from TMS building a journey profile which is then sent to ATO-OnBoard (ATO-OB) system. On the other hand, ATO-TS receives status reports and updated train information from ATO-OB, filters what is useful and forwards to TMS.

Plan Execution System (PES) also receives the Operational Plan from TMS and executes it, performing the traditional CTC functions. Specifically, it is PES which acts as a front-end towards TMS and connects to Trackside Protection System (TPS) subsystem which integrates Interlocking and ETCS systems. On the other side, PES receives updates from interlocking and ETCS systems and sends trains and infrastructures status changes to TMS, which uses these information to perform its functions such as re-estimating trains forecast, keep the Operational Plan up to date, solve conflicts, react to disruptions, etc.

TMS <> CCS shall equally support the Plan Execution System (PES) of Traffic CS if this component acts as the only front-end of TMS or both PES and CCS ATO Trackside (ATO-TS) if the TCS architecture will add the latter as a TMS front-end too. Furthermore, it should enable the connection of existing CTC systems and associated interlockings via adapters. Accordingly, TMS <> CCS is the single interface between TMS and the CCS system. **[**SPT3TMS-9724 **]**

TMS <> CCS is a bi-directional interface. The downstream provides information from TMS to the CCS. The upstream closes the feedback loop from CCS back to TMS. TMS sends the operational plan in one single information flow to both PES and ATO-TS.. TMS receives information from ATO-TS and PES in separate information flows. **[**SPT3TMS-16860 **]**

This concept contains the description of the abstract concepts of TMS <> CCS, a proposal for the messages to be exchanged, and the data model intended for the implementation of TMS <> CCS. **[**SPT3TMS-9721 **]**

4.2 Design Principles

4.2.1 Abstract concepts

4.2.1.1 Operational Plan

An Operational Plan is defined as the result of the planning process performed by the Planning System. It describes either a planned Operational Movement, Operational Restriction, or Operational Warning Measure through a temporal sequence of Operational Events to be implemented by SubSys ATO Trackside and/or Plan Execution System in the Area of Control configured for TMS. **[**SPT3TMS-16104 **]**

4.2.1.2 Operational Plan Operational Movement

An Operational Plan for an Operational Movement describes a Train Run by a precisely defined ordered set of Events. An Operational Movement is a train run of a specific train, at defined times, along a defined track path within the TMS Area of Control. **[**SPT3TMS-16105 **]**

4.2.1.3 Operational Plan Operational Restriction

An Operational Plan for an Operational Restriction contains a precisely defined ordered set of Operational Events and information on Operational Restrictions in an Operational Restriction Area. An Operational Restriction defines the parameters for the implementation of Operational Restriction Area(s) and the Operational Restriction(s) therein. **[**SPT3TMS-16100 **]**

As part of the Operational Plan, Operational Restrictions shall be planned by CMS subsystems and sent to TMS by the CMS/TMS interface which will be defined with a dedicated interface specification. In case of a disruption, which implies an unplanned restriction to be set up quickly, TMS shall do that and shall align CMS in case the effects of the disruption may have an effect on the time horizon managed by CMS. **[**SPT3TMS-16143 **]**

4.2.1.4 Operational Plan Operational Warning Measure

An Operational Plan for an Operational Warning Measure contains a precisely defined ordered set of Operational Events and information on Warning Areas from an operational perspective. An Operational Warning describes the temporal marking of a part of the railway network where approaching Physical Train Units triggers the activation of a Warning System. This is usually required on or near construction sites to protect Trackside Workers from dangerous situations. **[**SPT3TMS-16106 **]**

As part of the Operational Plan, Operational Warning Measures shall be planned by CMS subsystems and sent to TMS by the CMS/TMS interface which will be defined with a dedicated interface specification. It is assumed that every intervention of a maintenance team on site can be always planned in advance by CMS, even with a short or very short advice and therefore it is

not envisaged that an unplanned Warning _Measure may be defined by TMS and sent back to CMS for alignment. [SPT3TMS-16142]

The message types and data model for TMS <> CCS as described in this document guided by the design principles defined here after.

4.2.2 Every movement of a physical train unit shall be planned with an operational plan operational movement

For effective schedule optimisation TMS needs to know and control every movement and every track usage. To use the tracks in the most efficient way, the TMS creates operational plans for operational movements and observes their execution.

A Movement includes the following (list not exhaustive):

- Passenger train movements
- Freight train movements
- Shunting
- Train Maneuvers
- Yellow Fleet

[SPT3TMS-16103]

4.2.3 An operational plan movement defines exactly one train run

The granularity “one Train Run = one Operational Plan Operational Movement” has been chosen for several reasons: Firstly, it represents a train run that is a complete/self-contained entity, meaning with this that it contains all information necessary to execute the planned train trip from the beginning to the end of its planned path. Secondly, the data of an Operational Plan needs to be consistent and can be checked on this level; as the operational movement describes the full train trip, the consistency of its structure can be checked and validated easily without accessing any other data. Thirdly, because of runtime efficiency: if a train run changes, only the directly and indirectly affected Operational Plans shall be recalculated and redistributed. This reduces data volumes exchanged and avoids the risk of overwhelming data consumers (ATO-TS and PES). As a consequence, running many trains simultaneously means executing many Operational Plans in parallel. [SPT3TMS-9726]

A TMS is designed to interface one or more CCS systems, according to the configured Area of Control. An Operational Plan defines the whole train run, then also across the CCS borders; this is the preferred choice which is considered better than splitting the train run among the several involved CCS (if more than one), because avoids any reworking in case of changes of CCS multiplicity, or CCS borders adjustments, etc. [SPT3TMS-16108]

For cross-border trains, which run beyond the TMS area of Control, the Operational Plan defines the whole train run until the TMS border (national border if the TMS has a national scope). The same Operational Plan is sent to all the CCS systems concerned. Each CCS system knows its own map data and can therefore execute the corresponding part of the Operational Plan. [SPT3TMS-16107]

4.2.4 Dependencies between operational plans are ensured by movement event times, which are not changed by PES at any time

There are dependencies between operational movements (e.g., connecting trains or the defined order in which trains should pass a given location). The concept of movement event times is an explicit way of defining such dependencies: indeed, each movement event is characterized by timing information and a movement event time gives the actual time assigned to a movement event. A deeper view of this and other concepts related to operational movement are given when dealing with the Operational Plan Execution (Chapter 7). **[**SPT3TMS-9730 **]**

Additionally such dependencies can be established by links between events, which should be considered for example for legacy systems or if deemed mandatory. The order of the operational movements resulting from these dependencies is not changed by the Plan Execution System at any time but is the sole responsibility of TMS. However, manual intervention is possible in the event of an incident via CCS. Operational changes originating in CCS will be passed through TMS <> CCS to the TMS. **[**SPT3TMS-16109 **]**

4.2.5 The interface TMS <> CCS treats operational plans as data objects in a dynamic environment

Operational plans will be recalculated and redistributed based on the operational situation and the execution progress. This affects operational plans already in execution as well as those to be carried out in the future. The interface TMS <> CCS and the CCS components connected with it will be able to handle versions of operational plans from a technical and a business perspective. From a technical point of view, corresponding data fields must be kept available for versioning of operational plans. From a business point of view, it must be ensured that updates of operational plans are executable and fit with the current operational situation. **[**SPT3TMS-9729 **]**

To keep the amount of data manageable for all systems, the operational plan is split in separate planning objects (operational movements, operational restrictions, warning areas). The planning objects can be found in chapter 7.1.1 and Figure 12. These planning objects are separate units to reduce the amount of data. **[**SPT3TMS-9728 **]**

4.2.6 Abstraction Level and product independent interface definition

TMS <> CCS defines a common abstract interface that allows communication between TMS and CCS by defining the optimal set of data that the two systems must exchange, without the mention of a concrete software solution product. The specific choice of products is out of scope of the document.
[SPT3TMS-9727 **]**

4.2.7 Support of different migration scenarios

For reasons of migration and exchangeability TMS <> CCS shall support the following capabilities:

- TMS <> CCS shall be able to support two independent logical communication channels to Plan Execution System and to ATO Trackside. The two communication channels are defined by two different configurations of the TMS <> CCS.
- TMS <> CCS shall support the use of existing Centralized Traffic Control (CTC) systems and associated interlockings via adapters.
- TMS <> CCS shall support ATO grades of automation as well as Connected Driver Advisory Systems (C-DAS), with the potential use of adapters.

[SPT3TMS-9734 **]**

4.2.8 Support future updates of connected systems (Upgradeability)

TMS <> CCS interface shall support updates of the connected systems by including mechanisms to backwards compatibility between connected systems. **[**SPT3TMS-9733 **]**

4.2.9 Support of Publish-Subscribe-Principle

This TMS <> CCS interface specification doesn't force any choice about the coimmunication protocol, as it is purely focusing hte application layer. Nonetheless, it is recommended to rely on the Integration Layer communication platform, specified and defined by previous Shift2Rail initiative, and TMS/CCS data model defined by Transversal SP Domain. **[**SPT3TMS-16660 **]**

This seems the most logical choice which takes advantage from the studies and achievements already performed and which led to a common shared architecture at EU level. For the specification of this infrastructure, please see the related Shift2Rail documentation package. **[**SPT3TMS-16659 **]**

The choice of the most suitable communication infrastructure shall be performed at a later stage, and shall be taken by all relevant stakeholders in charge of defining the overall system architecture. **[**SPT3TMS-16661 **]**

Whichever is the final choice of the communication infrastructure is is assumed that the Publish-Subscribe-Principle is adopted. **[**SPT3TMS-16662 **]**

The communication model is based on a Publish-Subscribe-Principle to distribute data from one system to other involving system(s). Thereby, the leading distributing system acts as a producer and sends the data to a message broker. The message broker will dispatch the data to all subscribing systems, based on a routing key.

The communication is topic-based, enabling further systems to subscribe to the data distribution, when applicable.

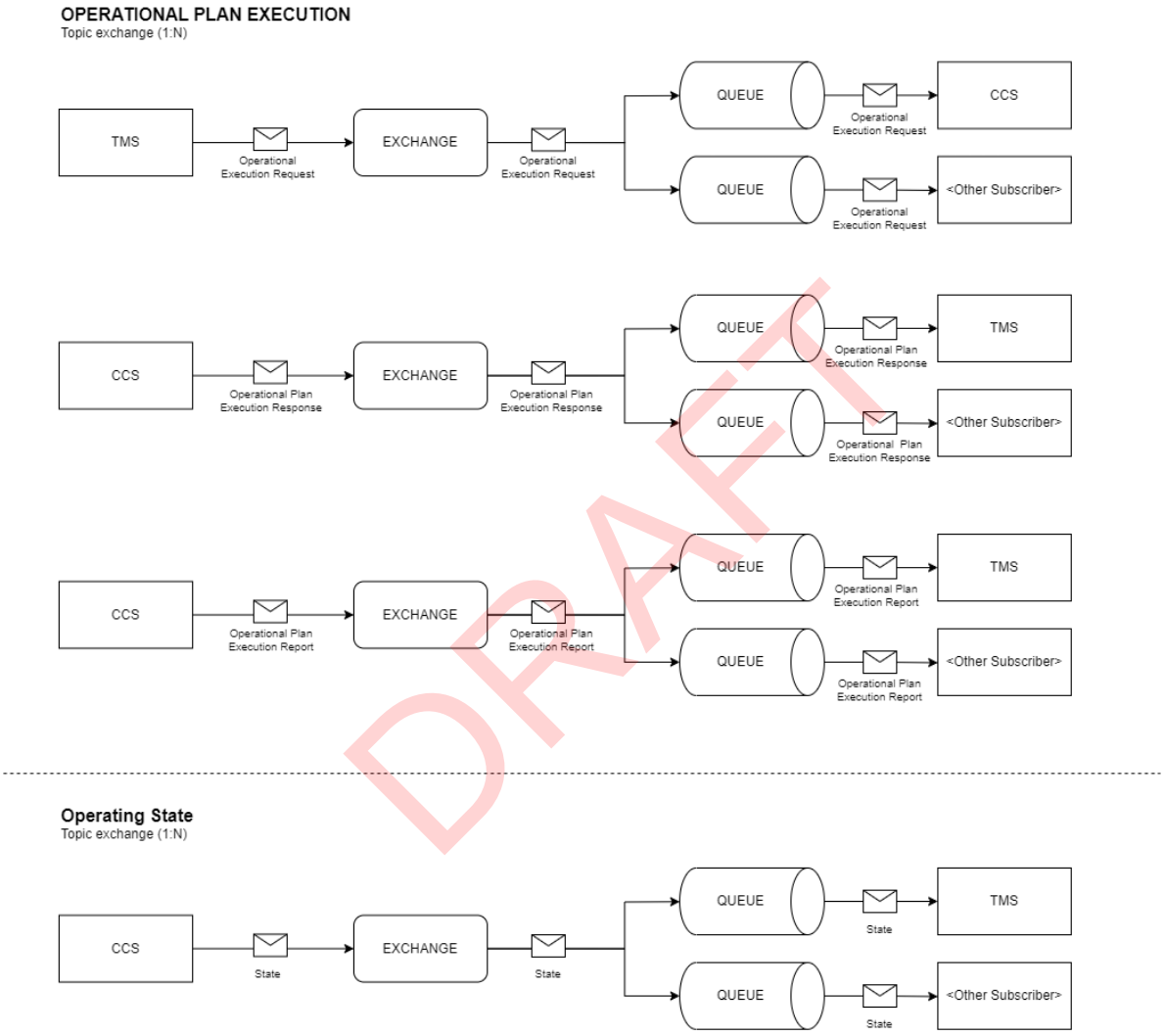
The following table illustrates the data distribution using Publish-Subscribe-Principle:

Use case:	**Producer:**	**Subscriber:**
Operational Plan Execution Request (Downstream)	TMS	CCS
		<Other Subscriber>
Operational Plan Execution Response (Upstream)	CCS	TMS
		<Other Subscriber>
Operational Plan Execution Report (Upstream)	CCS	TMS
		<Other Subscriber>

Use case:	Producer:	Subscriber:
Operating State (Upstream)	CCS	TMS
		<Other Subscriber>

[SPT3TMS-16665]

The following figure illustrates the data distribution using Publish-Subscribe-Principle:



[SPT3TMS-15916]

Figure 2: Data distribution according to Publish-Subscribe-Principle

s

4.2.10 Support two different configurations of TMS <> CSS

The TMS <> CCS shall support two different configurations in order to support the two independent logical communication channels from TMS to Plan Execution System and to ATO Trackside.

The following table describes the message flow:

Message:	Producer:	Subscriber:
Downstream:		
Operational Plan Execution Request (Operational Plan Movement)	TMS	Plan Execution System
Operational Plan Execution Request (Operational Plan Restriction)	TMS	Plan Execution System
Operational Plan Execution Request (Operational Plan Warning Measure)	TMS	Plan Execution System
Operational Plan Execution Request (Operational Plan Movement)	TMS	ATO Trackside
Upstream:		
Operational Plan Execution Response	Plan Execution System	TMS
Operational Plan Execution Report	Plan Execution System	TMS
Train Unit Report PE	Plan Execution System	TMS
Track Occupation State	Plan Execution System	TMS
Switchable Trackside Asset State	Plan Execution System	TMS
Restriction Area State	Plan Execution System	TMS
Warning Area State	Plan Execution System	TMS
Operational Plan Execution Response	ATO Trackside	TMS
Operational Plan Execution Report	ATO Trackside	TMS
Train Unit Report ATO Status	ATO Trackside	TMS
Train Unit Report ATO Capability	ATO Trackside	TMS

[SPT3TMS-15936 **]**

4.2.11 Data synchronization

It is assumed that the distributed systems TMS and CCS use asynchronous message transmission and are highly available, fault-tolerant and robust.
The TMS CCS Interface shall therefore implement functionalities that guarantee following requirements:

- It shall be possible to recognise that messages have been lost.
- It shall be possible to redeliver undelivered messages as soon as this is possible or retrieve them from the communication infrastructure. The quality of service should be ‘at least one-time delivery’
- It shall be possible to synchronize messages and data (e.g. after a system start or restart)

These requirements must be analysed for all interfaces to be specified with the System Pillar Reference Architecture.
It is therefore assumed that the required functionalities are defined system-wide for all relevant interfaces and that the resulting specification relates to the integration layer of each interface. **[**SPT3TMS-16094 **]**

4.2.12 Naming Convention for Enums

The naming convention of enums follows the "camelcase" convention in accordance to RailML/ RSM/Eulynx. **[**SPT3TMS-13914 **]**

4.2.13 Declarative indices in Json Schema

Declarative indices in Json Schema follows the Protobuf-Specification where class-ids are one-based and enums zero-based. This applies to all indices including "intId" in Json Schema. **[**SPT3TMS-15909 **]**

4.2.14 Naming and structure of Attributes

The naming and structures of attributes of the specification follows existing protocols. (e.g. SS-126) **[**SPT3TMS-13965 **]**

4.2.15 Time definition in Operational Plan

The use of offset rather than specific timestamps to reduce possible inconsistency and reduce the payload. **[**SPT3TMS-13975 **]**

4.2.16 Movement Restrictions for a specific train run are not considered as Timings of Movement Events

The explicit declaration of Movement Restrictions of an Operational Plan Movement for a specific train run is one of the aspect to be considered by PES to request the required flank protection type (track-enforced or non track-enforced flank protection) over a specific section on the railway network. **[**SPT3TMS-13927 **]**

PES on the other hand will always adhere to the Movement Restriction also if this means that the defined Movement Events cannot be accomplish in time. **[**SPT3TMS-16922 **]**

4.2.17 Train-change (due to damage) to a different train will result in an additional Operational Plan provision

In case, the train cannot realize the accepted Operational Plan due to damage, the train may be replaced by a different train. However, this will result in an additional Operational Plan. **[**SPT3TMS-13967 **]**

4.3 TMS <> CCS in the System Pillars (SP) Architecture

Infrastructure managers (IM) typically run separate processes for railway traffic planning and railway traffic execution. The interface TMS <> CCS connects planning and execution processes and thus external, IM-specific systems with CCS. **[**SPT3TMS-9736 **]**

TMS is for CCS an external, IM-specific system and not part of the CCS. TMS communicates over the interface TMS <> CCS with CCS. The CCS components Plan Execution System and ATO Trackside implement the TMS <> CCS interface. **[**SPT3TMS-9735 **]**

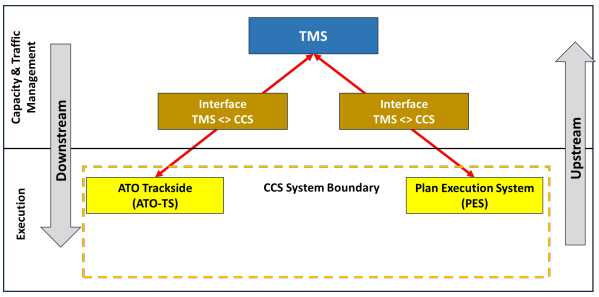
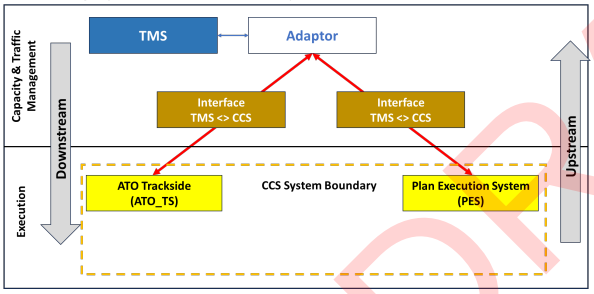
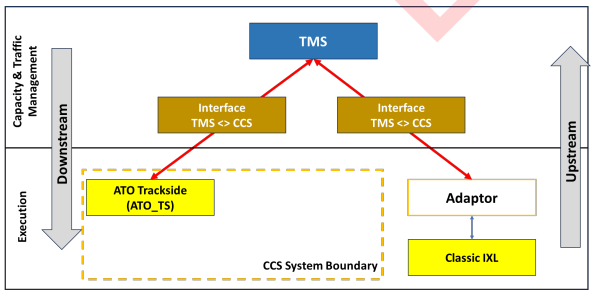
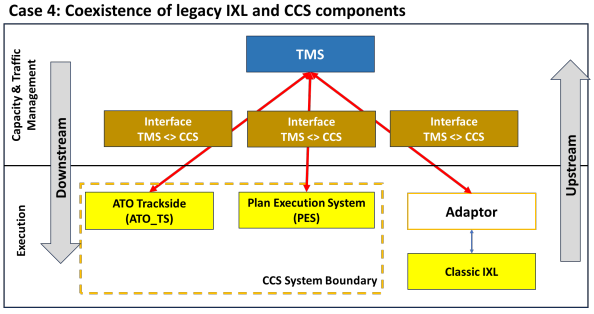
Based on the information taken from a received operational plan, Plan Execution System calculates and triggers requests for Drive Protection Section Groups, Movement Permissions, Usage Restriction Areas and Warning Areas via SCI-CMD in order to implement Operational Movements as well as Operational Plan Restrictions and Operational Plan Warning Measures. ATO Trackside generates instructions for ATO based on the information taken from the Operational Plan, so that Train Units autonomously drive the Operational Movements as planned. ATO Trackside and Plan Execution System report the execution progress of the Operational Plan back to TMS via the Operational Plan Execution Report and the Operating State. **[**SPT3TMS-9737 **]**

Since ATO Trackside and Plan Execution System are part of CCS, they are to be considered standard sub systems from an TMS perspective. **[**SPT3TMS-9741 **]**

4.4 Role of TMS <> CCS During Migration

TMS <> CCS plays a crucial role during any migration to new CCS systems. It bridges different technologies or development states and creates interoperability between the involved systems. The following table shows different blueprints for how TMS <> CCS could be applied. TMS <> CCS provides the flexibility to connect TMS to CCS and allows for connections with legacy interlockings (IXL). **[**SPT3TMS-9740 **]**

Table 1: Use cases of the application of the TMS <> CCS:

	This is the desired target state. A TMS that implements TMS <> CCS can be connected to CCS.
Case 2: Legacy TMS connected via adaptor to CCS 	If a legacy TMS cannot directly provide TMS <> CCS, an adaptor could “translate” from existing TMS interfaces to TMS <> CCS. This allows for existing TMSs currently used by various IMs to remain operational and interface to CCS.
Case 3: Legacy IXL fed by TMS <> CCS compatible TMS via adaptor 	This case is primarily thought as a migration step: First TMS <> CCS would be introduced between the TMS and the existing CTC/IXL by implementing an adaptor, that translates the Operational Plan to the language, which the CTC/IXL understands. In this case, it would be possible to already use ATO Trackside.
Case 4: Coexistence of legacy IXL and CCS components 	During migration, CCS infrastructure may be rolled out in steps. This means that parts of the infrastructure will be equipped with CCS components, whereas at the same time, others won’t. In this case, the TMS will exchange data via TMS <> CCS with CCS components and to an existing CTC/IXL via an adaptor.

[SPT3TMS-16862 **]**

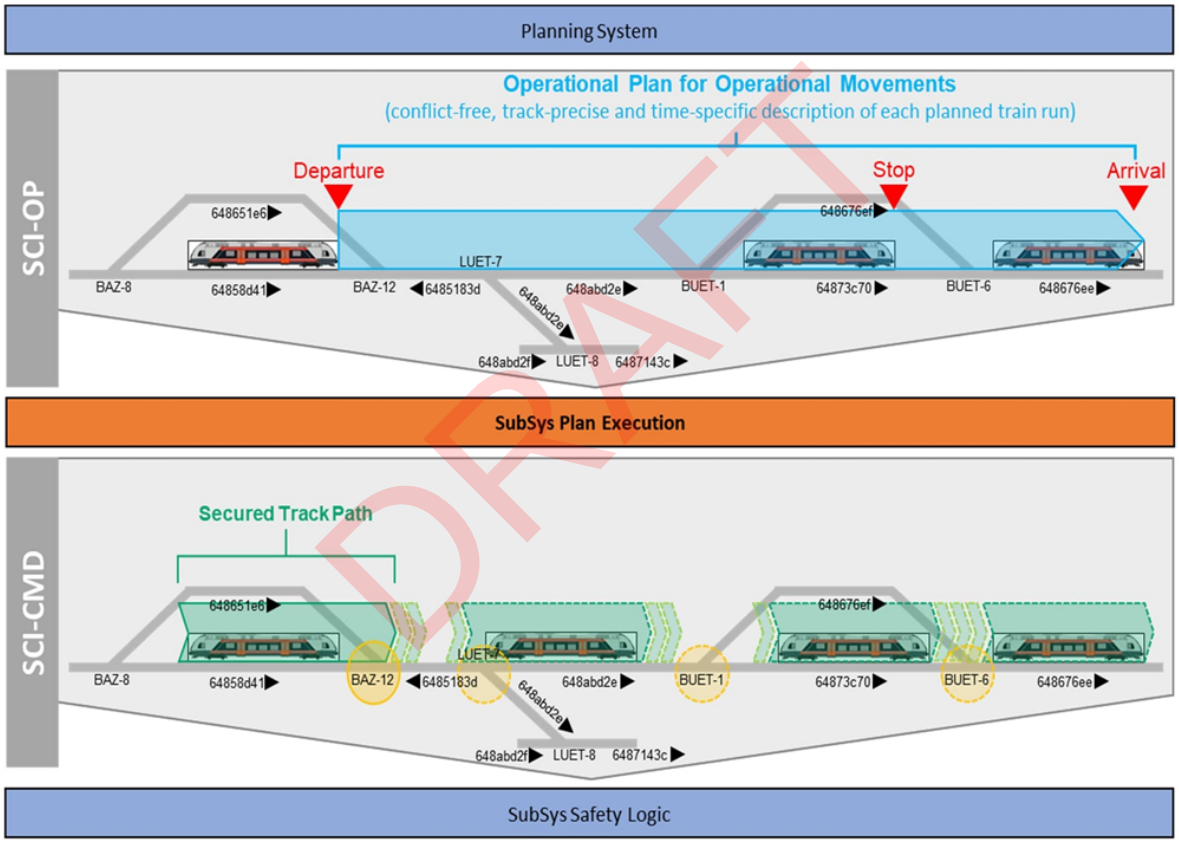
Note: The allocation of one or more TMS to an CCS as well as the spatial delimitation of the responsibilities of sub-systems shall not be discussed here. How the Areas of Control of the respective building blocks of CCS are to be intersected can only be determined at a later stage, when the scalability and performance of the sub-systems can be assessed.

5 CCS Sub-System “Plan Execution System”

Plan Execution System (PES) is a connecting sub-system between the TMS and CCS’ Safety Logic, managed by the TPS subsystem.

TMS manages the optimal capacity utilisation of the railway infrastructure in the AoC. TMS continuously re-plans the timetable and derives measures to optimise the flow of traffic on the network. TMS is typically a large IT system landscape which delivers an operational plan. The operational plan formed by the TMS is analysed and executed by PES subsystem. The Plan Execution System forms the discrete requests for securing the Track Path and sends them to the Safety Logic SL. **[**SPT3TMS-12493 **]**

5.1 Scope



[SPT3TMS-12495 **]**

Figure 3 Illustration of Plan Execution System (Plan Execution) and the two interfaces TMS <> CCS and SCI-CMD

Plan Execution System is a railway control and monitoring system. Its core functionality is the automatic and precise execution of the Operational Plans sent by the external TMS, see Figure 2.

- PES implements operational movements by timely requesting Movement Permissions for Physical Train Units and state changes of Switchable Trackside Assets for the drive ability of the railway network from the CCS Safety Logic.
- PES implements Operational Restriction Areas and Operational Warning Areas by timely requesting these as Usage Restriction Area and Warning Areas from Safety Logic.

- PES considers the dependencies between different Operational Plans as specified by TMS.

[SPT3TMS-12490 **]**

All of this core functionality is based on the knowledge of the TPS Operating State, a safe, logical representation of the actual state of railway operations in the Area of Control. Via the Standard Communication Interface Command (SCI-CMD), the Safety Logic provides the TPS Operating State. The TPS Operating State is then processed by PES and extended with additional information not known by TPS (references to Operational Plans, Operational Usage Restrictions, etc.). Afterwards, PES provides the Operating State with the additional information via (TMS <> CCS) to TMS, which is located outside the system boundary for CCS. **[**SPT3TMS-12489 **]**

PES makes a decisive contribution to CCS so that the overall system can benefit from new technical possibilities such as precise localisation and integrity check of Physical Train Units, the standardised control of the switchable trackside assets and the geometric safety logic of the CCS sub-system Safety Logic. **[**SPT3TMS-12492 **]**

PES's characteristics include:

- operates on abstract representations of real-world elements and objects
- operates in real-time
- works without knowledge of business rules (e.g. Timetable Rules, Capacity Planning) which are handled by the TMS andexpressed in the parameter values of Operational Plans and requests
- provides functionalities independent of the availability of TMS (manual input)

[SPT3TMS-12491 **]**

5.1.1 Features

- Precise implementation of Operational Plans for Operational Plan Movement, Operational Plan Restriction and Operational Plan Warning Measure sent by the TMS.
    - Requests the required drivability of the desired Track Path by sending some time in advance time requests to the Safety Logic for each switchable trackside asset, based on the operational situation.
    - Requests a Movement Permission for safe operational movement by sending just-in-time request to the Safety Logic with the optimal characteristics, based on the operational situation.
    - Implement Operational Restriction Areas (create, update, delete) by deriving required Usage Restriction Areas and sending requests to the Safety Logic at the specified time.
    - Implement Operational Warning Areas (create, update, delete) by sending requests to the CCS sub-system Safety Logic at the specified time.
    - Uses the unified data model defined in CCS that represents the railway network.
- Provides information about the execution progress of Operational Plans to the TMS.
- Processes information about the TPS Operating State and provide this information to the TMS in near real time.
- Provide information and commands needed for manual operation of PES via Workbench (temporary fall-back level for the TMS, unplanned manual interactions, use of existing CTC systems if no TMS is available during migration).

PES is characterized by very high availability, very low latency and very short and deterministic reaction times

5.1.2 Overview of functionality

PES and TPS provide the following functionality:

- **Operational Plan management**
  - Function: Process Operational Plan

[SPT3TMS-16128 **]**

- **Automatic plan execution**
  - **Drivability management**
    - Function: Observe drivability of railway network

Note: This is done by processing the TPS Operating State received via SCI-CMD **[**SPT3TMS-12502 **]**

- Function: Provide drivability of railway network

Note: This is done by providing information about drivability of railway network as part of the PES Operating State via TMS <> CCS and SWI-PE **[**SPT3TMS-12514 **]**

- Function: Control drivability of railway network

Note: This is done by sending the required commands to sub-system SL via SCI-CMD

- Subfunction: Calculate Trigger Points for drivability requests

[SPT3TMS-12512 **]**

- **Operational Movement management**
  - Function: Observe Operational Movement of Physical Train Units

Note: This is done by processing the TPS Operating State received via SCI-CMD **[**SPT3TMS-12511 **]**

- Function: Provide Operational Movement of Physical Train Units

Note: This is done by providing information about Operational Movement of Physical Train Units as part of the PES Operating State via TMS <> CCS and SWI-PE **[**SPT3TMS-12510 **]**

- Function: Control Operational Movement of Physical Train Units

Note: This is done by sending the required commands to the CCS sub-system SL via SCI-CMD

- Subfunction: Calculate MP
  - Subfunction: Calculate MP Extent
  - Subfunction: Calculate Risk Buffer
  - Subfunction: Calculate Risk Paths
- Subfunction: Calculate Trigger Points for MP requests

[SPT3TMS-12517 **]**

- **Operational Restriction management**
  - Function: Observe Usage Restriction Areas on railway network

Note: This is done by processing the TPS Operating State received via SCI-CMD **[**SPT3TMS-12516 **]**

- Function: Provide Operational Restriction Areas on railway network

Note: This is done by providing information about Operational Restriction Areas on railway network as part of the Traffic CS Operating State via TMS <> CCS and SWI-PE **[**SPT3TMS-12519 **]**

- Function: Control Operational Restriction Areas on railway network

Note: This is done by sending the required commands to SL via SCI-CMD
Subfunction: Define safeguard measures (Usage Restriction Areas) **[**SPT3TMS-12524 **]**

- **Operational Plan Warning Measure management**
  - Function: Observe Warning Areas on railway network

Note: This is done by processing the TPS Operating State received via SCI-CMD **[**SPT3TMS-12523 **]**

- Function: Provide Operational Warning Areas on railway network

Note: This is done by providing information about Operational Warning Areas on railway network as part of the PES Operating State via TMS <> CCS and SWI-PE **[**SPT3TMS-12522 **]**

- Function: Control Operational Warning Areas on railway network

Note: This is done by sending the required commands to SL via SCI-CMD **[**SPT3TMS-12521 **]**

- **Manual operation (via Workbench)**
  - Function: Manual plan execution
  - Function: Manual control of Switchable Trackside Assets

[SPT3TMS-12528 **]**

- **Device and Configuration Management**
  - Function: Import Configuration Data
  - Function: Activate Map Data

[SPT3TMS-12527 **]**

- **Monitoring and Diagnostics**
  - Function: Send diagnostics data

[SPT3TMS-12526 **]**

Besides the functionality listed above, PES provides the following functions to implement RAMSS requirements:

- Function: Use authentication and authorisation services
- Function: Support different System States, Operational States and Modes of Operation
- Function Determine System- / Operational State
- Function: Provide and receive System- / Operational State
- Function: Calculate integrity and check internal data model
- Function: Synchronies the state of Configuration Data and Operational Data

[SPT3TMS-12525 **]**

5.2 Out of Scope

The functions listed below are not part of the scope of Plan Execution System, although they are included in current systems such as existing centralised traffic control (CTC) or interlocking systems that are to be replaced by Trackside Protection System (TPS). Some of the listed functions have no relevance anymore in the CCS context, or they are in the scope of other systems such as the Planning System. The list is not exhaustive and may still change due to the adapted architecture. **[**SPT3TMS-12470 **]**

5.2.1 Operational Plan management

- PES does not support Operational Plans that offer different variants for the given Track Path from the departure to the arrival position. Plan Execution System only supports track-exact Operational Plans.
- PES does not resolve conflicts between different Operational Plans, nor does it change the order of Operational Movements dictated by the Operational Plans, both aspects are the responsibility of the TMS.
- PES does not perform route compatibility checks (catenary equipment, axle load compatibility, etc.) nor will it check whether the vehicle data specified in an Operational Plan for Operational Movement corresponds to the vehicle data reported by Safety Layer. These checks are not-safety related and must be done by the TMS or by the Railway Operating Company (ROC) before the Operational Plan for Operational Movement is sent to PES. The Plan Execution System will implement the Operational Plan although there might be incompatibilities between the properties of the planned train unit and the physical train unit or between the properties of the planned / physical train unit and the planned Track Path. It is intended that all safety-related checks are in charge of TPS component of TCS.

[SPT3TMS-12474 **]**

5.2.2 Drivability management

- Plan Execution System does not handle any manually or locally operated Switchable Trackside Asset, which do not have a technical interface to TPS. This means that e.g., manually operated points or level crossings in the Area of Control must either be replaced by automated ones or must be handled by IM specific operational processes.
- Plan Execution System does not support monitoring and control of catenary sections. Catenary sections must continue to be supported by the responsible systems outside of CCS.

[SPT3TMS-12472 **]**

5.2.3 Operational Movement management

- **Railway network deadlock detection**
  - Plan Execution System does not provide any kind of deadlock detection to prevent overfilling of the railway network, this functionality must be provided by the TMS.
- **Shunting**
  - Plan Execution System does not distinguish train runs and shunting movements. Any movement of a Physical Train Unit is either planned and processed as Operational Movement in an Operational Plan or is not supported. An Operational Movement is defined in terms of passages through relevant locations along with the associated arrival/departure/transit datetime. The "granularity" of the trip may be more or less detailed according tot he number of configured timing points and locations, but no refernce to field objects is given. TMS states where a train shouldbe and when, and with this level of abstraction, shunting movements can be treated as other movements.

[SPT3TMS-12478 **]**

5.2.4 Operational Plan Warning Measure management

- **Warning**
- Plan Execution System does not track or process the current position, properties and state of non-track bound vehicles, trackside personal or warning devices. According to the required safety level, the tracking and processing of such vehicles, personal and devices will be in the responsible of the TPS Safety Manager.
- Plan Execution System is not responsible for matching warning devices to Operational Warning Areas or setting up warning devices.
- Plan Execution System does not activate or de-activate warning devices to warn Physical Train Units, non-track-bound vehicles or trackside persons within or close to a warning area (PES does only create, update and delete warning areas).

[SPT3TMS-12476 **]**

5.2.5 Alarms regarding hazardous situations

PES may process alarms regarding hazardous situations (e.g avalanche sensors, hot box detector, short circuit in catenary sections, emergency call of driver, etc) in case it is stated more efficient for the user, otherwise these alarms shall be collected by a specific diagnostic system which will shall centralize and manage all alarms. The final decision shall be drawn as parte of the Traffic CS design. **[**SPT3TMS-12481 **]**

5.3 Context

5.3.1 CCS

The context of Plan Execution System within CCS (see Figure 2) is defined as follows:

- PES is a part of CCS. In terms of the overall CCS, it is a sub-system.
- Plan Execution System is defined as a product.
- Plan Execution System is currently envisaged as a non-safety-relevant system.

- The development process should nevertheless follow EN 50126-1 and EN 50128, as this is required from Basic Integrity level on. The actual safety requirements will be determined on the basis of the required risk analysis.
- As part of CCS, the Plan Execution System is intended for international use by European railway Infrastructure Managers.
- Plan Execution System is specified synchronously with other sub-systems such as TPS in the CCS and is connected to them via defined interfaces.
- With the TMS <> CCS, Plan Execution System provides the interface that defines the system boundary from CCS to the TMS.
- For the development of Plan Execution System and its interfaces compliance with international standards (e.g., TSI TAF/TAP, RailML) is observed.
- Plan Execution System is designed as a highly available system.
- Plan Execution System is operated together with exactly one logical instance of TMS and exactly one logical instance of TPS. (However, TMS should be able to operate multiple logical CCS instances simultaneously to divide the entire operational area of TMS into multiple Areas of Control each handled by a single CCS, thus achieving scalability).

[SPT3TMS-12485 **]**

Plan Execution System has an interface (SHI-PE) to other neighbouring PES instances to work in a joint network (as an example, via this interface parts of the PES Operating State can be exchanged with a neighbouring PES instance). **[**SPT3TMS-12484 **]**

5.3.2 Operation and training

- For the operation of Plan Execution System, an operating concept must be available.
  - The operating concept defines responsibilities and procedures between all organisations involved.
  - Service Level Agreements define which services are provided and by whom.
  - All measures and solutions affecting operation must be coordinated with the licensee and laid down in the operating concept.

[SPT3TMS-12482 **]**

- In case of detected errors or failures during operation, the state must be recorded.
  - It is recorded when an error occurs, when the error was eliminated and since when PES has been in operation again.

[SPT3TMS-12488 **]**

- Before handing over the system, the system operator and the personnel must be trained in the parts of the system relevant to them.

[SPT3TMS-12487 **]**

5.3.3 Organisation

- Plan Execution System supports generic concepts and can therefore be used independently of the operational organisation or exact operational use within the limits of the scope of the system.

[SPT3TMS-12501 **]**

- Plan Execution System shall not have any additional and specific restrictive effects on the organisation, neither by the possible number of workplaces, their local distribution across

sites or distribution in a site, nor by the operational concept (administration, monitoring, maintenance).

[SPT3TMS-12500]

5.3.4 Legal Aspects of Concern

1. Occupational safety e.g., for usability graphical user interfaces
2. Functional safety ("safety")
3. Information security ("security")
4. Product liability, e.g., due to traceability aspects

Plan Execution System, as other CCS sub-systems, is intended for international use. A reference to the legal basis will be made as soon as the countries in which Plan Execution System is used are known. [SPT3TMS-12497]

5.4 Environment

5.4.1 Physical Influences

The technical system Plan Execution System is executed on a computing platform containing hardware as well as software (board support package, operating systems, runtime environment, etc.). It will be in the responsibility of the computing platform operator to deal with physical influences. [SPT3TMS-12505]

5.4.2 System Interfaces

According to the current Traffic Control and Supervision system (see Traffic CS System Concept specification), Plan Execution System can be connected to other subsystems outside of the CCS system boundaries. Specifically, PES is connected to TMS through TMS <> CCS interface, whose specification is the goal of the present document and is an evolution of the “SCI-OP” interface specification described in the Reference CCS Architecture (RCA). [SPT3TMS-16126]

6 CCS Sub-System “ATO Trackside”

As described in the previous section, nowadays modern TMS systems are more and more large IT systems, which perform a complex and advanced set of railway applications, to improve and optimize traffic regulation and optimization of jurisdiction areas getting larger and larger, according to a roadmap which aims to encompass large regional areas, whole Countries and even more at European level; this supports the more general goal to build a Single European Railway Area (SERA). **[**SPT3TMS-16019 **]**

Inside the overall Railway architecture conceived by System Pillars initiative, the ATO Trackside is a connecting subsystem between the TMS and the ATO On-Board (ATO-OB), which it is external to the Traffic Control and Supervision System interfacing TMS; ATO Trackside subsystem can interface one or more ATO-OB subsystems. **[**SPT3TMS-16022 **]**

The ATO Trackside sub-system plays a key role, connecting to TMS, re-elaborating the Operational Plan received by TMS for every train inside its controlled area and building its associated Journey Profile (JP). The Journey Profile of as train is a concept which has been conceived and developed in GoA2 and described in subset-125; while the Operational Plan describes the train trip in terms of stations, remarkable locations and timing points crossed during its trip, linked paths among them and sets constraints for arrival, departure and dwell times, the Journey Profile details the journey as passing through a sequence of Segment Profiles (SPs), including additional timing information and temporary infrastructure information, derived by the Operational Restrictions received by TMS. **[**SPT3TMS-16021 **]**

Both TMS and ATO Trackside share the same data model and representation of the infrastructure, built in advance, stored into the Digital Register (DR) and published when deemed necessary. **[**SPT3TMS-16018 **]**

6.1 Scope

The core functionality of ATO Trackside subsystem is the automatic and precise transaction and execution of the Operational Plan data sent by the TMS: **[**SPT3TMS-16028 **]**

- ATO Trackside subsystem timely transacts operational movements defined in the Operational Plan towards the ATO infrastructure and ATO-OB for execution.
- ATO Trackside subsystem transacts ATO-OB status reports back to TMS via train unit report

[SPT3TMS-16030 **]**

As a connecting subsystem between TMS and subsystem ATO-OB, the ATO Trackside subsystem gives a decisive contribution to SP architecture so that the overall railway system can benefit from new technical possibilities of ATO railway operation. **[**SPT3TMS-16026 **]**

The coordination with the processing of the Plan Execution System (PES) and the following downstream systems is assured, by using the same Operational Plan instance sent by TMS

and the synchronisation of the execution onboard. ATO Trackside is based on the exported constraint that PES subsystem and the following downstream systems fulfil the requirement to secure any train movement at any time.
ATO Trackside characteristics include: [SPT3TMS-16025]

- Operates on abstract representations of real-world elements and objects
- Operates in real-time
- Works without knowledge of Timetable Rules, which are handled by the TMS and are expressed in the parameter values of Operational Plans

[SPT3TMS-16027]

6.1.1 Features

ATO Trackside shall provide the following features: [SPT3TMS-16032]

- Receive and transform TMS Operational Plan data into the ATO specific format of Journey Profile (JP), to provide JP to the ATO On Board subsystem
    1. Management of JP transmission status and local storage of JPs

[SPT3TMS-16132]

- Maintain ATO Map Data and related alignment with the reference data model adopted by all other cooperating subsystems, which means
    1. Preloading of new map data and check of the integrity
    2. Activation of the previously loaded Map Data

[SPT3TMS-16134]

- Handle the Operational Plan received by TMS in relation to the valid Map Data
    1. Validity check of the Operational Plan received by TMS
    2. Conversion of Operational Plan into the Subset-126 format for ATO-OB use
    3. Conversion of Reports from ATO-OB into TMS-CCS related data format for TMS use

[SPT3TMS-16133]

- Receive and transform status changes of ATO communication sessions between an ATO-OB and ATO Trackside subsystem and provide this information as Train Unit Report to TMS.

[SPT3TMS-16130]

- Receive and transform ATO Status Report information according to Subset-126 of an ATO-OB and provide this information as Train Unit Report to TMS.

[SPT3TMS-16135]

- Acquire detailed information of the ATO Trackside subsystem status including all involved hardware and software components and interfaces and provide diagnosis and status data to a centralized diagnostic system according to SP architecture. This includes the logging of all interface status and communication data available.

[SPT3TMS-16138]

- Enable remote operation for configuration and maintenance.

[SPT3TMS-16137]

- Enable identity management as part of security processes, according to the definition of Secure Component given by SP Cybersecurity Secure Component Specification.

[SPT3TMS-16136]

ATO Trackside subsystem is characterized by very high availability, very low latency and very short and deterministic reaction times. [SPT3TMS-16035]

6.2 Out of Scope

According to the design of SP architecture, the ATO Trackside subsystem is not part of the safety related railway operation system chain. Therefore, this specification doesn't deal with any safety related function or characteristic of a railway system.
Furthermore, according to definition of the SP overall architecture, the TMS and the functions listed in this chapter are not part of the scope of ATO Trackside subsystem. The list is not exhaustive and may still change due to the evolutions and refinements on-going as part of SP design. [SPT3TMS-16034]

6.2.1 Operational Plan management

As it regards the execution of the Operational Plan: [SPT3TMS-16037]

- ATO Trackside subsystem does not resolve conflicts between different Operational Plans, nor does it change the order of Operational Movements dictated by the Operational Plans, both aspects are the responsibility of the TMS, only.

[SPT3TMS-16141]

- ATO Trackside subsystem does not support Operational Plan that offer different variants for the given Track Path from the departure to the arrival position. ATO Trackside subsystem only supports track-exact Operational Plans.

[SPT3TMS-16140]

- ATO Trackside subsystem does not check the Operational Plan against compatibility nor vehicle data nor other analogue data. However, ATO Trackside subsystem does perform a validity check of Operational Plan data against the valid Map Data stored in ATO Trackside subsystem.

[SPT3TMS-16139]

6.2.2 Communication and management of ATO-onBoard

ATO Trackside subsystem does not provide ATO infrastructure data to other subsystems defined in SP architecture. ATO Trackside subsystem only uses the Map Data provided at configuration time, including the ATO specific data, like validity check of the specific Operational Plan data,

conversion and/or adaptation of Operational Plan data or conversion of ATO-OB reports into the data model designed for the TMS-CCS interface specification. [SPT3TMS-16043]

6.2.3 Management of ATO-onBoard

ATO Trackside subsystem does not manage single vehicles or ATO-OB units; this is a function performed by ATO-OB, which is a system external to Traffic Control and Supervision System, which ATO Trackside system is a part of. [SPT3TMS-16045]

6.3 Context

6.3.1 CCS

The context of ATO Trackside within CCS is defined as follows: [SPT3TMS-16041]

- ATO Trackside is part of CCS in SP architecture. In terms of the overall CCS, it is a sub system
- ATO Trackside is actually envisaged as a non-safety-relevant system.
- The development process should nevertheless follow EN 50126-1 and EN 50128, as this is required from Basic Integrity level on. The actual safety requirements will be determined based on the required risk analysis.
- As part of CCS the ATO Trackside is intended for international use by European railway Infrastructure Managers.
- ATO Trackside is specified synchronously with other subsystems such as TPS in the CCS and is connected to adjacent subsystems via defined interfaces.
- With the TMS <> CCS, ATO Trackside provides the interface that defines one system boundary from CCS to the TMS.
- For the development of ATO Trackside, and its interfaces the compliance with international standards (as Subset-125, TSI TAF/TAP) is respected.
- ATO Trackside is designed as a highly available system.
- ATO Trackside is operated together with exactly one logical instance of a Traffic Control and Supervision (TCS).
- A TCS is designed to configure and manage exactly one logical instance of a ATO Trackside .
- TMS shall be able to operate multiple TCS instances simultaneously to divide the entire operational area of TMS into multiple Areas of Control each managed by a single CCS system, thus achieving scalability.

[SPT3TMS-16050]

ATO Trackside has an interface (ATO Airgap) to the one or more (usually more) controlled ATO-OB instances, to manage a pool of trains (via this interface Journey Profiles and Train Capability Reports are exchanged). Through ATO airgap ATO-OB receives from ATO Trackside the journey profile for the specific train movements, while ATO-OB provides to ATO Trackside the ATO Status Report containing information about the train status as well as any update about relevant train characteristics, giving TMS the possibility to re-evaluate the train forecast, detection of possible arisen conflicts and the re-evaluation oft he operational plan. [SPT3TMS-16051]

6.3.2 Operation and training

The main goal of ATO Trackside is the ATO related technical communication and operational plan transformation into a journey profile between the TMS and the further ATO downstream systems. **[**SPT3TMS-16047 **]**

Therefore, there is no railway operation performed manually at the ATO Trackside. As a consequence, there will be no operational regulation nor training required for railway operator in relation to the ATO Trackside. **[**SPT3TMS-16046 **]**

Moreover, the ATO Trackside maintenance and configuration processes will be executed remotely via the CCS support features, as diagnostic, maintenance, identity and access management services. **[**SPT3TMS-16049 **]**

Of course, the regulation and training for maintenance and configuration of ATO Trackside must be provided to the responsible personnel of the CCS support for these centralized services. This is not in scope of this document. **[**SPT3TMS-16048 **]**

6.3.3 Organization

ATO Trackside supports generic concepts and can therefore be used independently of the operational organization or exact operational use within the limits of the scope of the system. ATO Trackside shall not have any additional and specific restrictive effects on the organisation, neither by the possible number of workplaces, their local distribution across sites or distribution in a site, nor by the operational concept (administration, monitoring, maintenance). **[**SPT3TMS-16057 **]**

6.3.4 Legal aspects of concern

1. Signalling safety (“safety”)
2. Information security (“security”)
3. Product liability, e.g. due to traceability aspects

[SPT3TMS-16066 **]**

ATO Trackside, as other CCS subsystems, is intended for international use. A reference to the legal basis will be made as soon as the countries in which ATO Trackside subsystem is used are known. **[**SPT3TMS-16063 **]**

6.4 Environment

The core functionality of ATO Trackside subsystem is the automatic and precise transaction and execution of the Operational Plan data sent by the TMS: **[**SPT3TMS-16061 **]**

6.4.1 Physical influences

ATO Trackside consists of hardware or virtual environment and software. The hardware component, if used, might be exposed to physical influences. It will be in the responsibility of the computing platform operator to deal with physical influences. **[**SPT3TMS-16060 **]**

There is no railway operation performed manually at the ATO Trackside , therefore there is also no physical input-output device dedicated to this subsystem. **[**SPT3TMS-16059 **]**

It is assumed that all maintenance and configuration tasks will be performed remotely by specific operators dealing with maintenance and diagnostic. **[**SPT3TMS-16058 **]**

6.4.2 System Interfaces

According to the current Traffic Control and Supervision system (see Traffic CS System Concept specification), ATO Trackside can be connected to several CCS subsystems and one or more instances of ATO-OB subsystem outside of the CCS system boundaries. In addition to these, ATO Trackside is connected to TMS through TMS <> CCS interface, whose specification is the goal of the present document and is an evolution of the “SCI-OP” interface specification described in the Reference CCS Architecture (RCA). **[**SPT3TMS-16056 **]**

7 General Structure of the Interface TMS <> CCS

TMS <> CCS is a bi-directional interface. Its main purpose is the provision of Operational Plans to CCS. This results in a single, but very significant downstream message, Operational Plan Execution Request. The upstream messages sent from CCS are required by the TMS to detect deviations between Operational Plans and their execution as well as to identify upcoming or existing conflicts between Operational Plans and to develop appropriate countermeasures. The upstream of the interface TMS <> CCS consists generally of the following streams:

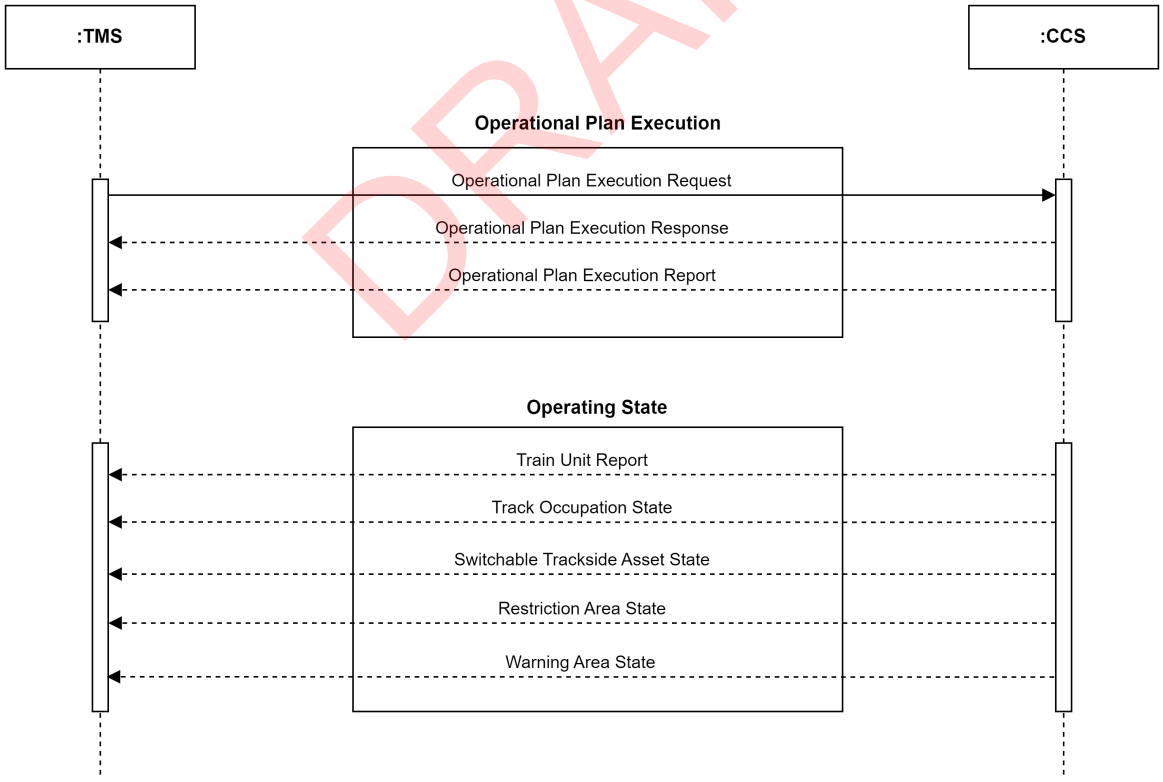
- Operational Plan Execution Response (acceptance or rejection of execution request)
- Operational Plan Execution Report (progress of an accepted Operational Plan)
- Operating State (actual state of railway operations

[SPT3TMS-9881 **]**

The messages transmitted via the interface TMS <> CCS are logically divided into two groups:

1. Operational Plan Execution
2. Operating State **[**SPT3TMS-9880 **]**

An overview on the main structure of TMS <> CCS is illustrated in Figure 3.



[SPT3TMS-6665 **]**

Figure 4 General structure of TMS <> CCS

Operational Plan Execution:
The main purpose of the TMS <> CCS is the provision of Operational Plans to CCS. Accordingly, the Operational Plan Execution comprises the major downstream message type from an external TMS to CCS as well as relevant upstream return messages from CCS to the TMS. All necessary abstract concepts, message type descriptions and further details of the Operational Plan Execution will be described in chapter 7. **[**SPT3TMS-9879 **]**

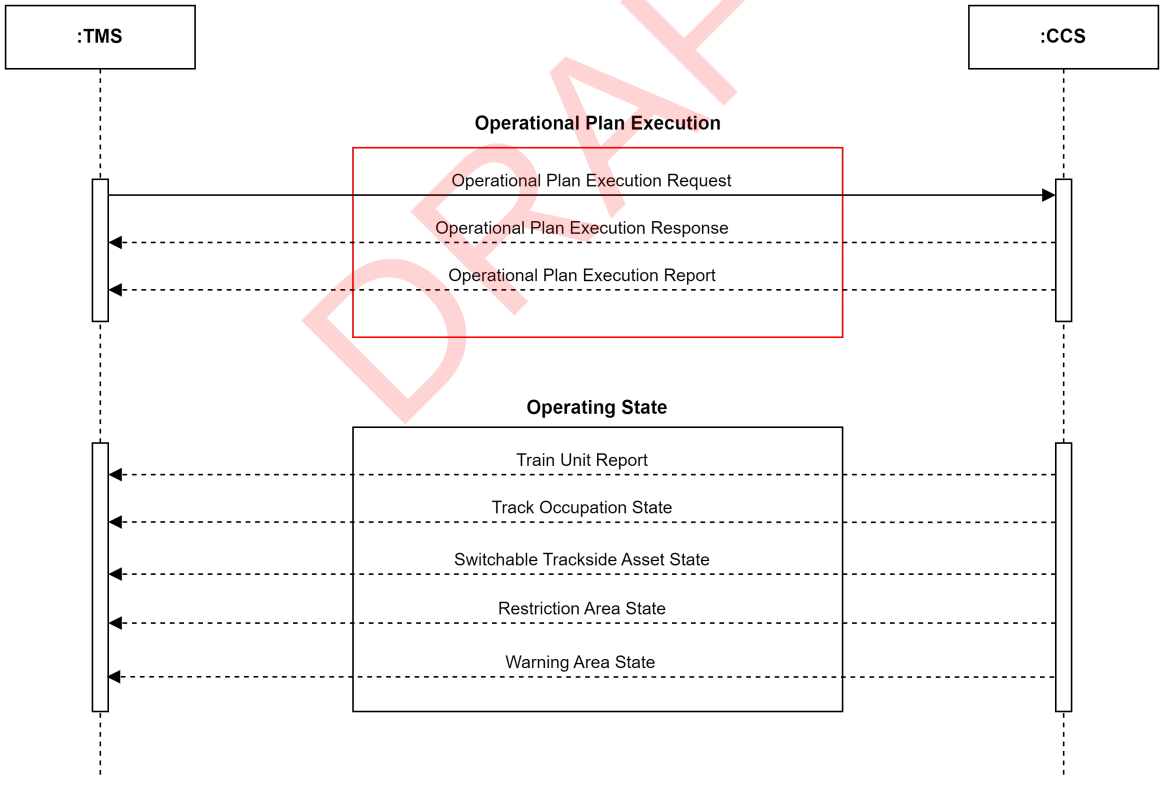
Operating State:
For providing a better picture on the current state of CCS to the external TMS, the Operating State delivers numerous upstream messages. They deliver, among others, information about the currently operating Train Units, the occupation of tracks, or the settings of the switchable trackside assets. All necessary abstract concepts, message type descriptions and further details of the Operating State will be described in chapter 8. Operating State . **[**SPT3TMS-9878 **]**

8 Operational Plan Execution

This chapter describes the abstract concepts as well as the messages of the Operational Plan Execution of TMS <> CCS. The messages of the Operational Plan Execution are embraced into three message streams, as illustrated in Figure 4.

This Chapter draws on RCA Terms and Abstract Concepts as per RCA.Doc.14, which is considered recommended supplementary reading. However, it must be stressed that this specification evolved towads the SP architecture and relies on SP architectural principles and defined systems and components. **[**SPT3TMS-9885 **]**

Abstract concept “Operational Plan Execution”:
The Operational Plan Execution is the logical representation of all messages concerning the execution of Operational Plans in the Area of Control. This includes the Operational Plan Execution Request, the Operational Plan Execution Response and the Operational Plan Execution Report. **[**SPT3TMS-9884 **]**



[SPT3TMS-6664 **]**

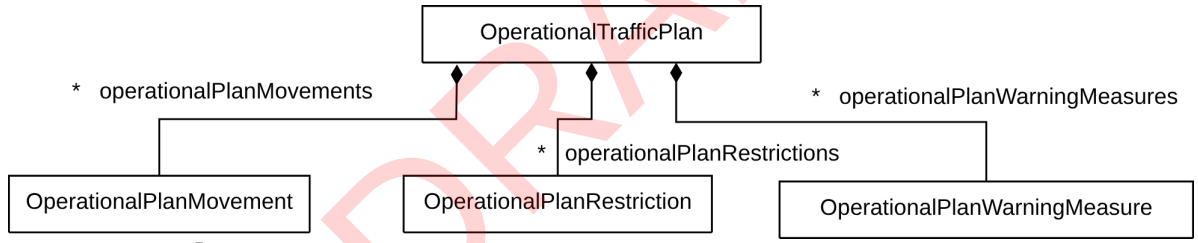
Figure 5 Message streams of TMS <> CCS with focus on the Operational Plan Execution

8.1 Operational Plan

Abstract Concept "Operational Plan":

The Operational Plan is the result of the planning process performed by TMS. It describes either a planned Operational Plan Movement, Operational Plan Restriction, or Operational Plan Warning Measure through a temporal sequence of Operational Events to be implemented by ATO Trackside and/or Plan Execution System in the Area of Control, as illustrated in Figure 5. **[**SPT3TMS-12141 **]**

An Operational Plan will be issued by TMS for any operationally relevant activity. This comprises all movements of Physical Train Units (incl. shunting operations), restrictions due to e.g., infrastructure maintenance and construction works, and warning measures during restrictions. The implementation of an Operational Plan can be requested by an Operational Plan Execution Request (see details 8.2 - Operational Plan Execution Request) that is issued by the TMS to CCS. CCS reacts to the Operational Plan Execution Request with Operational Plan Execution Response and Operational Plan Execution Report. An Operational Plan contains all necessary information for CCS PES and ATO Trackside to implement the Operational Plan. Plan Execution System will be enabled to provide the capacity on the infrastructure for executing the Operational Plan. ATO Trackside will be enabled to generate necessary driving instructions for the Physical Train Unit assigned to the respective Operational Plan. **[**SPT3TMS-13901 **]**



[SPT2TS-126806 **]**

Figure 6 General domain structure.

8.1.1 Operational Event

Abstract concept "Operational Event":

An Operational Event is a description of a single planned action (e.g. stop/passage of a Train Unit, start/ end of an Restriction Area, start/end of an Warning Area) defined in the Operational Plan.
[SPT3TMS-13951 **]**

Depending on the type of an Operational Plan, different Operational Events with different properties are required. Therefore, the Operational Plan distinguishes between:

- Movement Events
- Restriction Events
- Warning Measure Events

An Operational Event possesses constraints with respect to time and position at which it shall be executed. Accordingly, all actions necessary for the event to be implemented shall be carried out before that defined time. It is in the responsibility of the consuming Subsystem to ensure the timely triggering of necessary requests to ensure that the Operational Event will be implemented as planned.
[SPT3TMS-13953 **]**

8.1.2 Operational Plan Movement

Abstract concept "Movement":

A Movement is a train run of a certain train, at defined times, along a defined track path within the Area of Control. **[**SPT3TMS-13954 **]**

Abstract concept "Operational Plan Movement":

The Operational Plan Movement defines the parameters for the implementation of a Movement.

[SPT3TMS-6601 **]**

.

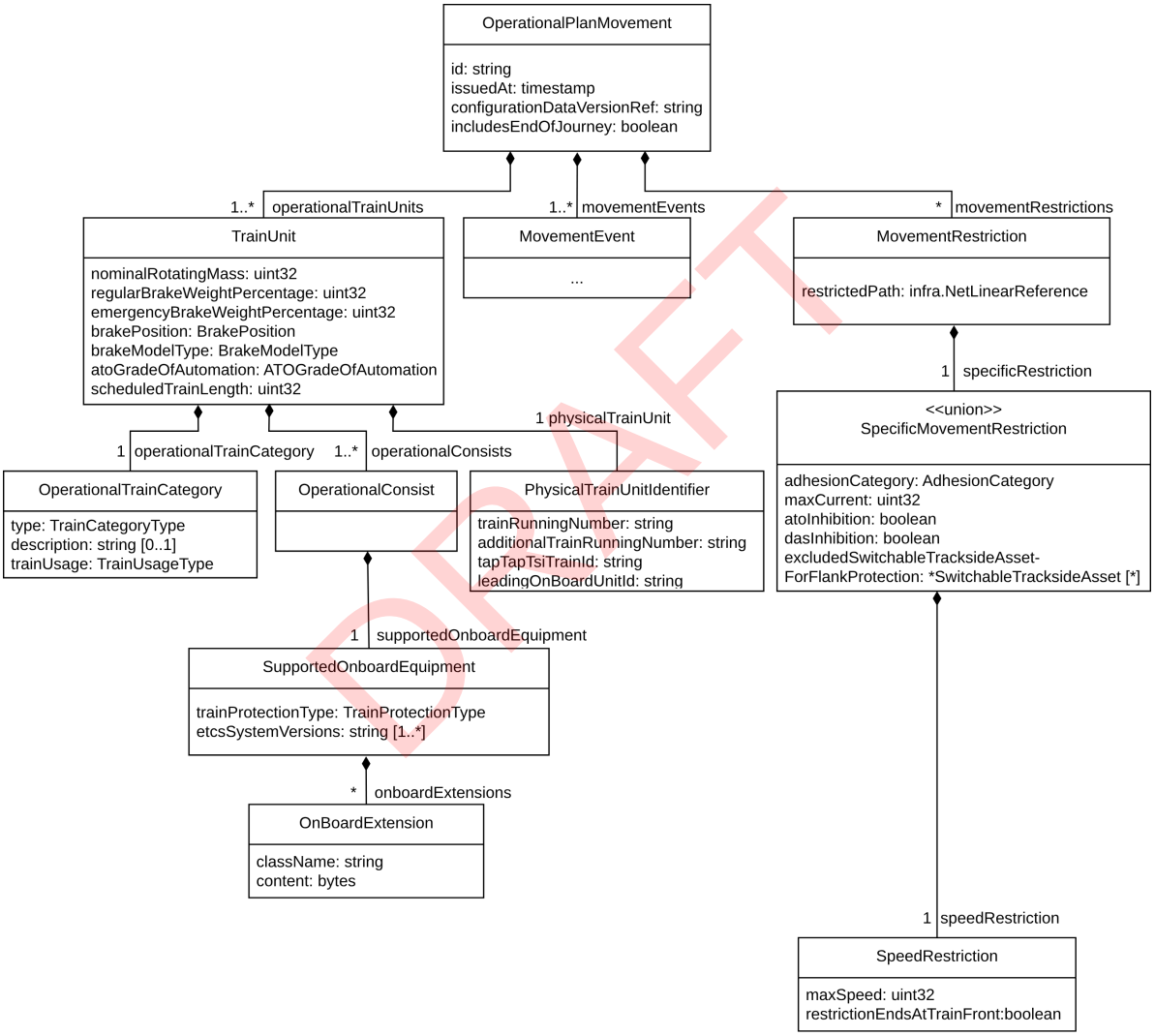


Figure 7 Class diagram for Operational Plan Movement

[SPT2TS-124418 **]**

8.1.2.1 Train Unit

Abstract concept: “Train Unit”:

The Train Unit describes the train unit to be operated with the Operational Plan Movement. It describes the train unit by a train unit identifier, operating parameters and a sequence of consists.
[SPT3TMS-12140 **]**

The Train Unit describes all relevant aspects of a train which may have a significant function to the operation. This can be categorized into the following:

- Mass & weight
- Braking behavior
- Grade of Automation

[SPT3TMS-13902 **]**

8.1.2.2 Operational Consist

Abstract concept: “Operational Consist”:

The Operational Consist describes a sequence of the Physical Vehicles and the operating parameters of the Physical Consist planned to be operated. **[**SPT3TMS-11222 **]**

The following describes cases (list is not exhaustive), where an Operational Consists becomes relevant:

- Adding Physical Consists or Physical Vehicles to a Physical Train Unit
- Changing the order of the Physical Consists of a Physical Train Unit
- Removing Physical Consists from a Physical Train Unit
- Direction change (alters the order of Physical Consists in the Physical Train Unit)
- Change of the Operational Train Category
- Change of the train number

[SPT3TMS-11221 **]**

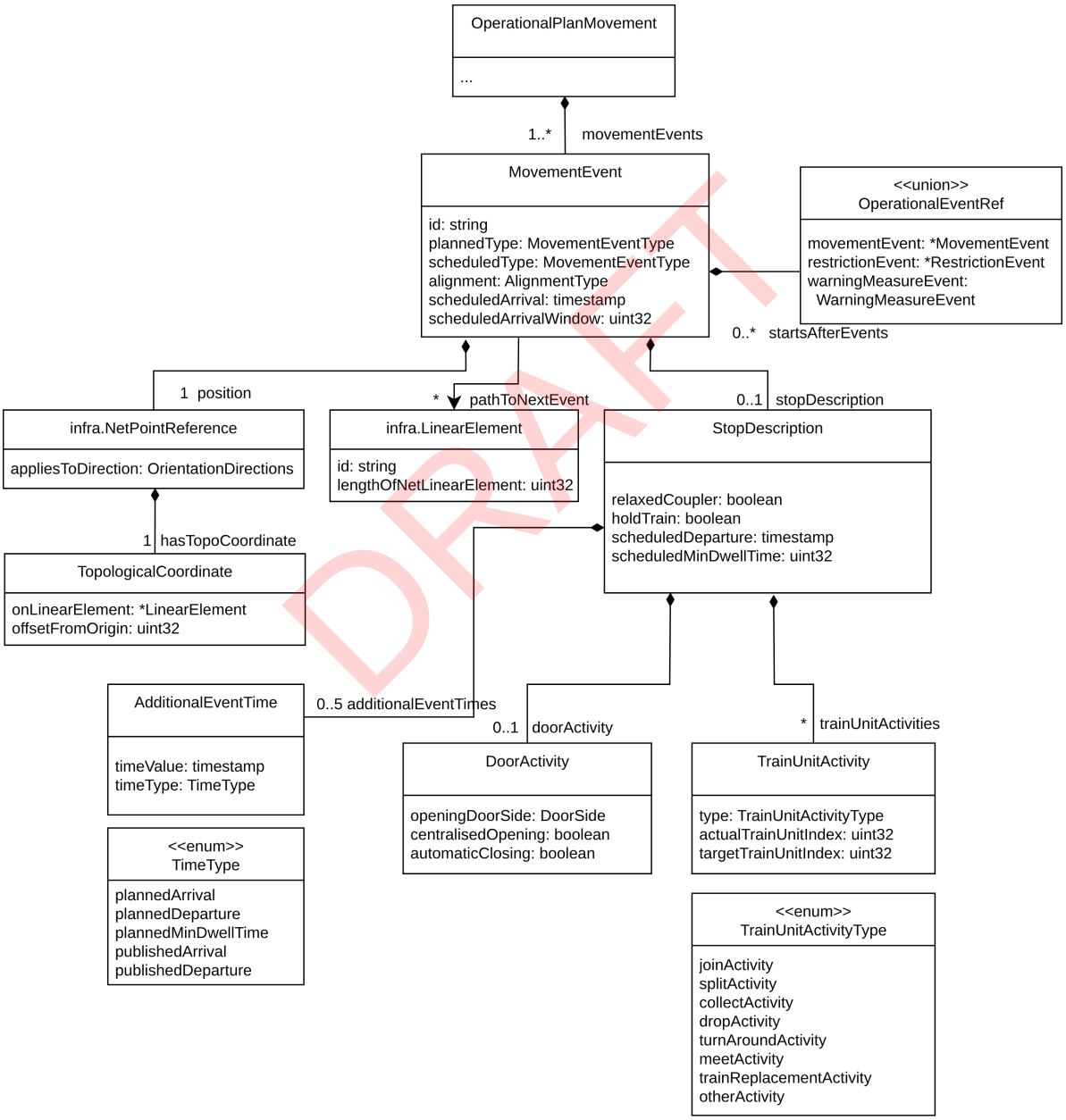
8.1.2.3 Movement Event

Abstract concept “Movement Event”:

Movement Events are events that are specified in the Operation Plan Movement. The list of Movement Events within an Operation Plan Movement describes the planned train journey, the train path used, all passages and stops as well as the specified times and the activities included.

[SPT3TMS-12143 **]**

Movement event describes activities the train implements at some specific position on the infrastructure (TrackEdge). **[**SPT2TS-124422 **]**



[SPT2TS-126809 **]**

Figure 8 : Class diagram for Movement Event

8.1.2.4 Event Link

Abstract concept “Event Link”:
May be relevant for both SERA compliant and legacy systems, which needs the connection to other Operational Plans instead of considering timing information. The Event Links are optional and used when deemed necessary .
[SPT3TMS-16878 **]**

8.1.2.5 Stop Description

Abstract concept “Stop Description”:
Characteristics and activities to be carried out at a planned stop of a Physical Train Unit.
A Stop Description describes the planned stop along a train journey which is relevant for Implementation and triggering of requests via SCI-CMD and AoE SS-126.

A Stop Description is categorized by the following:

- Timing
- Activity

[SPT3TMS-12142 **]**

8.1.2.6 Movement Event Time

Abstract concept “Movement Event Time”: [SPT3TMS-13942 **]**

A Movement Event Time provides the actual time assigned to a Movement Event. **[**SPT3TMS-13941 **]**

With each Movement Event, timing information is provided. A Movement Event can possess several types of timing information.

The Movement type determines how many types of timing information are required for it:

- **Timings for PASS:** scheduledArrival and scheduledArrivalWindow are required values.
- **Timings for STOP:** A Stop Movement can either possess an arrival, a departure, or both of a planned train run.

[SPT3TMS-13943 **]**

The following table describes the different Timing Points during a train run:

Time:	Description:	Usecase:
scheduledArrival	Latest scheduled arrival time	Relevant to express scheduled arrival time.
scheduledArrivalWindow	scheduled arrival window	Relevant for passages.

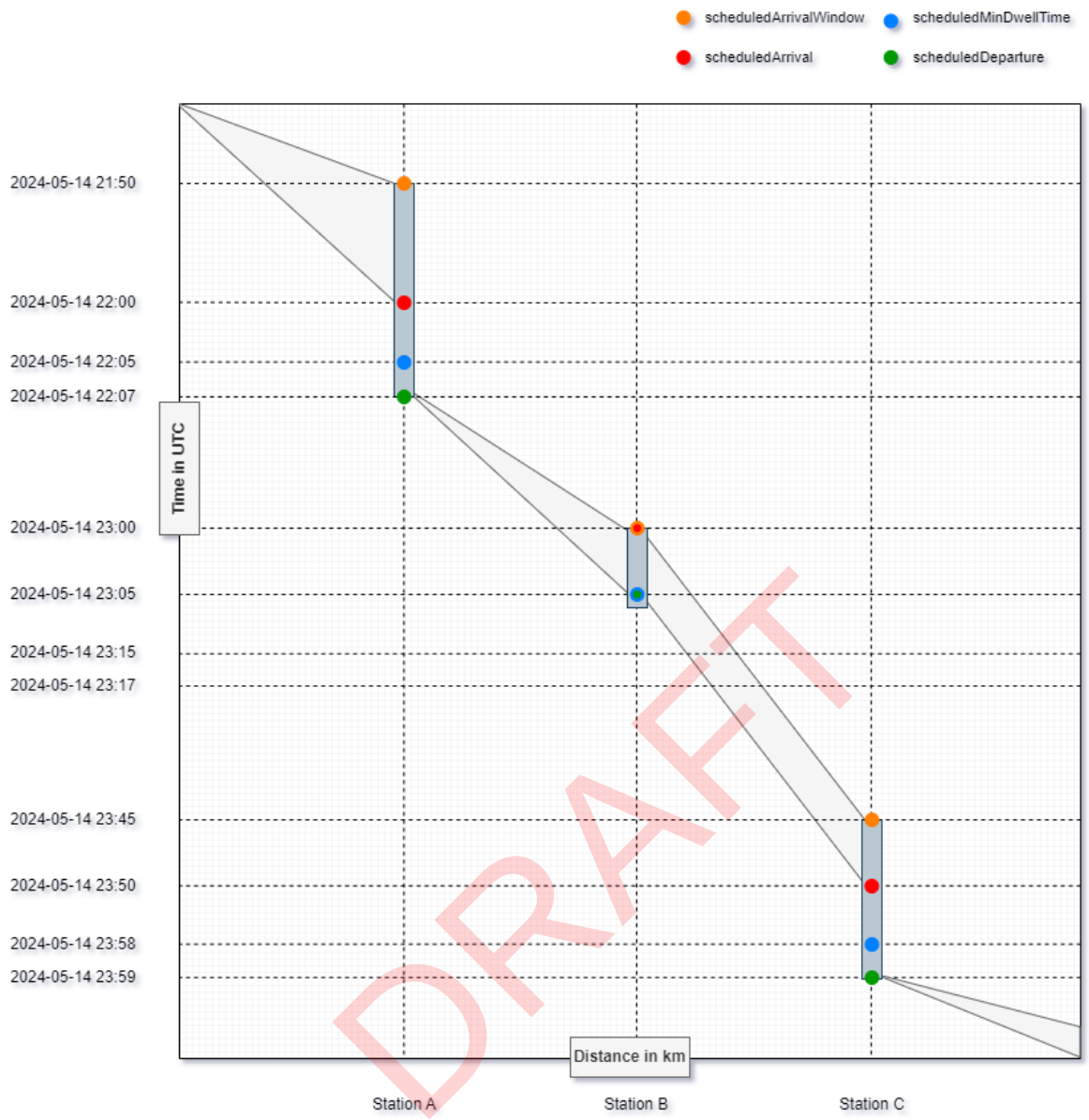
Time:	Description:	Usecase:
scheduledDeparture	Earliest scheduled departure time	Relevant to express scheduled departure time.
scheduledMinDwellTime	Minimum waiting time between arrival and departure time. Planning aspect and is not reflected in the dispatching. ARS will not consider this value as its logic. The value will be sent unchecked via SS-126 to ATO onboard.	Relevant to express scheduled waiting time.
plannedArrival	Planned arrival time, provided by CMS.	Relevant for C-DAS to display differences between planned and scheduledarrival time.
plannedDeparture	Planned departure time, provided by CMS.	Relevant for C-DAS to display differences between planned and scheduled departure time.
plannedMinDwellTime	Minimum waiting time between arrival and departure time. Planning aspect and is not reflected in the dispatching. ARS will not consider this value as its logic. The value will be sent unchecked via SS-126 to ATO onboard.	Relevant to express planned waiting time.
published Arrival	Published arrival time	Displayed arrival time for end customers online and on plattforms. May differ from the planned arrival time.
published Departure	Published arrival time	Displayed departure time for end customers online and on plattforms. May differ from the planned departure time.

[SPT3TMS-13944 **]**

Time ranges in Timing Points:

The time ranges are useful, because ATO cannot drive an exact timeline and would get permanently re-controlled by the TMS. Therefore, certain degrees of freedom should be provided for each Movement Event. They give the consuming Plan Execution System and ATO Trackside certain leeway in determining the precise processing of Events. **[**SPT3TMS-13938 **]**

The following diagram illustrates the timing points in context: **[**SPT3TMS-15473 **]**



[SPT3TMS-15472]

Figure 9: ZWL Diagram

Updating of Operational Plan Movement:

When updating an Operational Plan Movement via SCI-OP, TMS shall specify an Operational Plan Movement which starts with one or several completed Movement Event by Plan Execution System and ATO Trackside. This Movement Event works as a reference between the two versions and therefore helps to assure gap-free and unambiguous transition between two successive versions of the Operational Plan Movement. Subsequent Movement Events in the new version may differ from the previous version. [SPT3TMS-13935]

Example 1/3 of Movement Event Timings:
The following example describes an earlier Cargo train run incl. a passing.

Attribute:	Value:	Remark:
Published		
publishedArrival	null	Not relevant as cargo train timings are not published for passengers.
publishedDeparture	null	
Planned		
plannedArrival	4:30:00	plannedArrival and scheduledArrival differs, as the train is earlier as planned.
plannedDeparture	4:30:00	plannedDepartureand ScheduledDeparture differs, as the train is earlier as planned.
plannedMinDwellTime	0	plannedMinDwellTime and ScheduledDepartureWindow differs, as the train is earlier as planned.
Scheduled		
scheduledArrival	2:12:30	PASS
ScheduledDeparture	2:12:30	
ScheduledArrivalWindow	00:03:00	PASS

[SPT3TMS-15474 **]**

Example 2/3 of Movement Event Timings:
The following example describes an on-time passenger train run (S-BAHN) incl. a stop.

Attribute:	Value:	Remark:
Published		
publishedArrival	10:00:00	publishedDeparture differs from plannedDeparture by 30 sec. to streamline a punctual departure.
publishedDeparture	10:02:00	
Planned		
plannedArrival	10:00:00	plannedArrival and scheduledArrival matches, as the train is on time.
plannedDeparture	10:02:30	plannedDeparture and ScheduledDeparture matches, as the train is on time.
plannedMinDwellTime	0:01:00	-
Scheduled		

Attribute:	Value:	Remark:
scheduledArrival	10:00:00	-
scheduledDeparture	10:02:30	-

[SPT3TMS-15477]

Example 3/3 of Movement Event Timings:
The following example describes a delayed passenger train run (Intercity) incl. a stop.

Attribute:	Value:	Remark:
Published		
publishedArrival	09:56:00	publishedDeparture differs from plannedDeparture by 30 sec. to streamline a punctual departure.
publishedDeparture	10:04:00	
Planned		
plannedArrival	09:56:00	plannedArrival and scheduledArrival differs as the train is delayed.
plannedDeparture	10:04:30	plannedDeparture and ScheduledDeparture differs as the train is delayed.
plannedMinDwellTime	0:06:00	plannedMinDwellTime differs from ScheduledDepartureWindow as the train is delayed.
Scheduled		
scheduledArrival	10:03:00	-
scheduledDeparture	10:06:00	-

[SPT3TMS-15476]

8.1.2.7 Track Path

Abstract concept “Track Path”:

The Track Path is a gap-free and track-specific route on the railway network. It is used for describing the path of a Movement. **[**SPT3TMS-12147 **]**

8.1.2.8 Movement Restriction

Abstract concept “Movement Restriction”:

An Movement Restriction is an operational, non-safety-related restriction relevant for a Movement and specified within the Operational Plan Movement. **[**SPT3TMS-12153 **]**

Types of Movement Restrictions:

- Temporary speed restriction zones:
  - Use Case: By restricting the speed of specific train unit over a certain track section, TMS can influence the type of flank protection to be implemented by PES (“dynamic flank protection”). TMS uses this feature to optimize traffic flow.
    Note: This kind of speed restriction is not safety-related and it is train unit specific and therefore is not implemented as a Restriction Area with the Traffic Protection System (TPS)
- Temporary low adhesion zones:
  - Use Case: Used by ATO-TS to control acceleration and braking, e.g. when there are leaves on the rails or in frosty conditions
- Temporary restricted «Max current» zones:
  - Use Case: Used by ATO-TS, e.g. if there are many trains in a station and it is known that the power grid is not quite as strong; so, for certain trains, during a certain time, a limitation of the maximum power consumption can be imposed.
    Note: This is not the same as shifting the departure and arrival times of train units, which could also lead to lower power consumption at a given time.
- Temporary ATO inhibition zones
  - Use Case: If the train unit would run over a section that is not constantly monitored, or a Class B route.
  - Use Case: Occasionally to stimulate the driver's attention

[SPT3TMS-13955 **]**

8.1.2.9 Example of an Operational Plan Movement

The properties of the Movement described in this section are to be illustrated by means of an example. Therefore, the following Figure shows a simplified application of an Operational Plan. **[**SPT3TMS-11244 **]**

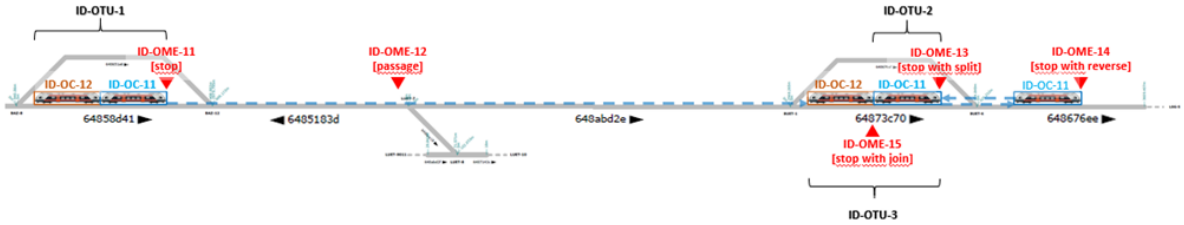


Figure 10 Train run

[SPT3TMS-11243 **]**

The depicted Physical Train Unit travels from left to right and its Operational Plan contains the following (simplified, not exhaustive) information:

· The Operational Train Unit (ID-OTU-1) which consists of 2 Operational Consists (ID-OC-11 at position 1 and ID-OC-12 at position 2) departs at the Movement Event (ID-OME-11) at Track Edge Point (64858d41).

· The Operational Train Unit passes the Movement Event (ID-OME-12) at Track Edge Point (6485183d).

· The Operational Train Unit stops at the Movement Event (ID-OME-13) at Track Edge Point (64873c70) and performs a split. A new Operational Train Unit (ID-OTU-2) is defined, which contains only the Operational Consist (ID-OC-11 at position 1).

· The Operational Train Unit stops at the Movement Event (ID-OME-14) at Track Edge Point (648676ee) and performs a reverse (in this case no new Operational Train Unit needs to be defined, because it's a single Operational Consist).

· The Operational Train Unit arrives at the Movement Event (ID-OME-15) at Track Edge Point (64873c70) and performs a join. A new Operational Train Unit (ID-OTU-3) is defined, which contains both Operational Consists at reversed order (ID-OC-12 at position 2 and ID-OC-11 at position 1).

A data file printout of the depicted example is provided in the Appendix (section 12.1).

[SPT3TMS-11245 **]**

8.1.3 Operational Plan Restriction

Abstract concept “Operational Plan Restriction”

The Operational Plan Restrictions defines the timing and spatial dimension of Restriction Area(s) and the Restriction(s) therein to be implemented. **[**SPT3TMS-11247 **]**

The Operational Plan Restriction is used to pursue maintenance, construction works, reduce speed in some areas. There can be either long-term or short-term planned restrictions which are provided by TMS as Operational Plan Restriction to the CCS; the CCS will create them as safety measures, as they are safety-related. **[**SPT3TMS-13956 **]**

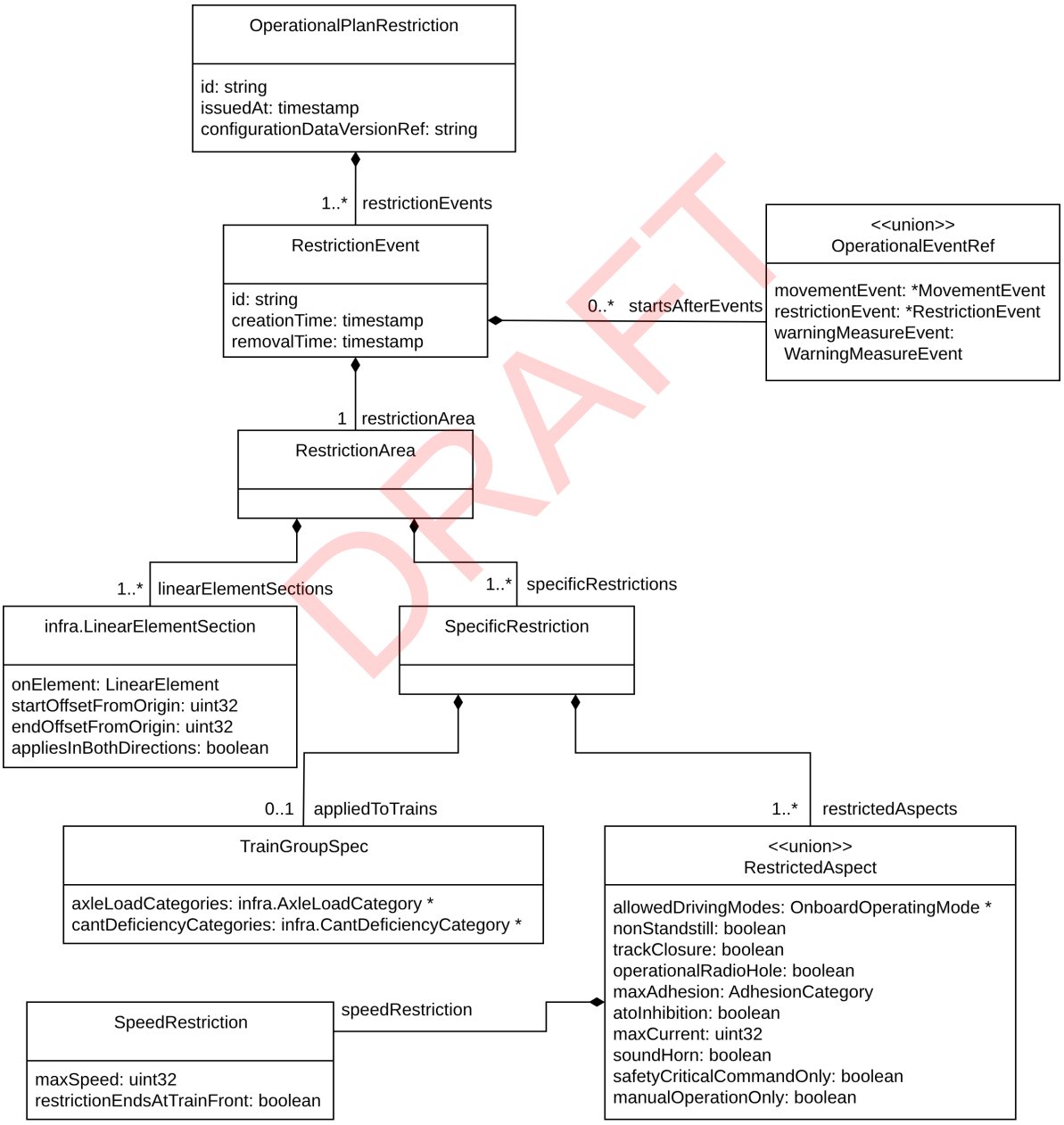


Figure 11 Class diagram for Operational Plan Restriction

[SPT2TS-124448 **]**

8.1.3.1 Restriction Area

Abstract concept “Restriction Area”:

The Restriction Area is an entity of the Operational Plan Restriction and the Operating State. It describes the spatial dimensions of a planned or already implemented Restriction within the Area of Control. The Restriction Area is an area of the railway network that is temporarily restricted for safety reasons. **[**SPT3TMS-11257 **]**

The spatial dimension of a Restriction Area is described as Track Edge Sections which provide the possibility to define a direction. Therefore, Operational Plan Restrictions which are valid in one or both directions of travel can be represented. A Restriction Area can contain one or more Track Edge Sections. One or more Operational Plan Restrictions are assigned to a Restriction Area. These are described in the following subsection. **[**SPT3TMS-11258 **]**

8.1.3.2 Restriction Event

Abstract concept “Restriction Event”:

Restriction Event is a transactional unit for activation and deactivation of a set of restrictions. **[**SPT3TMS-11263 **]**

Timing Information in Restriction Events:

Timing information at which the Restriction Event shall take place during execution of the Restriction are required. Unlike Movement Events there are no Degrees of Freedom foreseen for the timing points of Restriction Events. **[**SPT3TMS-11400 **]**

8.1.4 Operational Plan Warning Measure

The Operational Plan Warning Measure defines the timing and spatial dimension of of Operational Warning Area(s) and the Warning Measure(s) therein to be implemented. **[**SPT3TMS-11239 **]**

Figure 12 Class diagram for Warning measure

Maintenance activity: review this diagram with TCCS team to restore original link

An Operational Plan Warning Measure describes a type of warning measure within a Warning Area. Warning Measures are implemented by an Automatic Warning System which, in turn, consists of a set of collective Warning Devices and/or individual Warning Devices. The available Warning Measures depend on national/local safety regulations (defining, among others, early warning thresholds, warning duration, etc.), Warning Devices at hand, and their specific configuration. Operational Plan Warning Measures do not influence Movements.
[SPT3TMS-12149 **]**

8.1.4.1 Warning Area

Abstract concept “Warning Area”

The Warning Area is an entity of the Operational Plan Warning Measure and the Operating State. It describes the spatial dimensions of a planned or already implemented Warning Measure within the Area of Control. The Warning Area is an area on the railway network in which driving trains can potentially impose danger, to Authorised Trackside Persons (e.g., to infrastructure maintenance staff)
[SPT3TMS-11288 **]**

The spatial dimensions of a Warning Area are described as Track Edge Sections Track Edge Sections which provide the possibility to define a direction. Therefore, Operational Plan Warning Measures which are valid in one or both directions of travel can be represented. A Warning Area can contain one or more Track Edge Sections. One or more Operational Plan Warning Measures are assigned to a Warning Area. These are described in the following subsection.
[SPT3TMS-11287 **]**

8.1.4.2 Warning Device

Maintenance activity: This attribute is subject to change, depending on the final decision whether Warning Devices shall be integrated in CCS. **[**SPT3TMS-16931 **]**

Abstract concept "Warning Device":
Describes the warning devices **[**SPT3TMS-12151 **]**

List of planned warning devices assigned to the Warning Area that shall be "READY_TO_WARN_IN_OWA" for an Operational Plan Movement to enter the Warning Area. **[**SPT3TMS-13949 **]**

8.1.4.3 Warning Measure Event

Abstract concept "Warning Measure Event":
Warning Measure Event is a transactional unit for activation and deactivation of a set of warning measures. **[**SPT3TMS-11290 **]**

Timing Information in Warning Measure Events
Timing information at which the Restriction Event shall take place during execution of the Operational Plan Restriction are required. As for the Operation Movement Events there are Degrees of Freedom foreseen for the timing points of Warning Measure Events. **[**SPT3TMS-11292 **]**

8.2 Operational Plan Execution Request

Abstract concept “Operational Plan Execution Request”:
The Operational Plan Execution Request is the request to implement an Operational Plan. It is sent by the TMS to Plan Execution System and ATO Trackside via TMS <> CCS.
[SPT3TMS-15932 **]**

The following describes the TMS requirement for updating the
operational plan x minutes before the scheduling event: **[**SPT3TMS-15931 **]**

The TMS shall update the Operational_Plan one minute before execution at least **[**SPT3TMS-12239 **]**

The following subsection 7.1.1 contains the detailed explanation and abstract concepts used for the Operational Plan. The subsection 7.1.2 thereon holds the description of the message in detail.
Note: All the messages of the Operational Plan contain a header and a specific packet of the sending system. The header contains generic data, such as Ids, whereas the specific packet contains the actual information to be transferred. **[**SPT3TMS-15933 **]**

8.2.1 Message: Operational Plan Execution Request

Note: All the messages of the Operational Plan contain a specific packet of the sending system. The specific packet contains the actual information to be transferred. **[**SPT3TMS-9818 **]**

Description:
This message is sent by the TMS to request the execution of an Operational Plan (initial or update) in the CCS, in detail through Plan Execution System and/or ATO Trackside.
[SPT3TMS-11401 **]**

Attributes:
The structure of the message corresponds to: 10.1.1 - Abstract Class: OperationalPlan
[SPT3TMS-15927 **]**

8.3 Operational Plan Execution Response

Abstract concept “Operational Plan Execution Response”:

The Operational Plan Execution Response is the response (acceptance or rejection) to an Operational Plan Execution Request. It is provided by ATO Trackside and Plan Execution System via TMS <> CCS. **[**SPT3TMS-6669 **]**

The Operational Plan Execution Response is one of three upstream message types for reactions of the CCS on an Operational Plan Execution Request. It contains only one message: operational_plan_execution_response. **[**SPT3TMS-12263 **]**

The message is issued for the following types of Operational Plans:

- Operational Plan Movement (described in section 8.1.2 - Operational Plan Movement)
- Operational Plan Restriction (described in section 8.1.3 - Operational Plan Restriction)
- Operational Plan Warning Measure (described in section 8.1.4 - Operational Plan Warning Measure)

and and is provided by both CCS Plan Execution System and ATO Trackside. **[**SPT3TMS-9814 **]**

8.3.1 Message: Operational Plan Execution Response

Description:

This message is the response to an operational_plan_execution_request to report either the acceptance or the rejection of an Operational Plan. A rejection is only reported if the Operational Plan is not feasible. **[**SPT3TMS-9815 **]**

Attributes:

The structure of the message corresponds to: 10.5.1 - Class: ExecutionResponse **[**SPT3TMS-15928 **]**

8.4 Operational Plan Execution Report

Abstract concept “Operational Plan Execution Report”:

The Operational Plan Execution Report describes the execution progress of the Operational Plan. It is provided by ATO Trackside and Plan Execution System via TMS <> CCS. **[**SPT3TMS-6668 **]**

The Operational Plan Execution Report is one of three upstream message types for reactions of the CCS on an Operational Plan Execution Request. It contains only one message: operational_plan_execution_report. **[**SPT3TMS-12275 **]**

The message is issued for the following types:

- Movement (described in section 8.1.2 - Operational Plan Movement)
- Restriction (described in section 8.1.3 - Operational Plan Restriction)
- Warning Measure (described in section 8.1.4 - Operational Plan Warning Measure)

and and is provided by both CCS Plan Execution System and ATO Trackside. **[**SPT3TMS-12289 **]**

8.4.1 Message: Operational Plan Execution Report

Description:

This message reports an update on the execution state of a previously accepted Operational Plan. An operational_plan_execution_report shall be sent at each state change of an Operational Event for each accepted Operational Plan by the sub-systems responsible for its implementation. **[**SPT3TMS-9812 **]**

Each state change of an Operational Event issues a new Operational Plan Execution Report to the related Operational Plan. Accordingly, Operational Plan Execution Reports will be sent to the TMS based on discrete events but not otherwise, e.g., not by fixed time intervals. **[**SPT3TMS-13957 **]**

With the discrete updates of Operational Events, the Operational Plan Execution Report does not provide a real-time or near-time picture of the operational situation. More detailed (near-time) updates on the progress of an Operational Movement will be provided via dedicated messages of the Operating State, that is to say the Train Unit Report (cf. section 8.1), the Track Occupation (cf. section 8.1.3.4) and the Switchable Trackside Asset State (cf. section 8.2.2.2). Detailed updates on Operational Plan Restrictions will be provided via dedicated messages of Restriction Areas (cf. 8.3.1.2) and Warning Areas (cf. 8.4.2.2). **[**SPT3TMS-13958 **]**

The Operational Plan Execution Report will be returned by CCS sub-systems implementing the Operational Plans. The sub-systems will provide their own Operational Plan Execution Reports to the TMS and the TMS is responsible for combining the information provided together with the Operating State (see section 8), into a coherent image.
The Operational Plan Execution Report enables the ATO Trackside and Plan Execution System to inform the TMS about the state of an Operational Event, as well as existing failures or warning concerning this Operational Event.
An explanation about the data amount is given in chapter 3.2. **[**SPT3TMS-13959 **]**

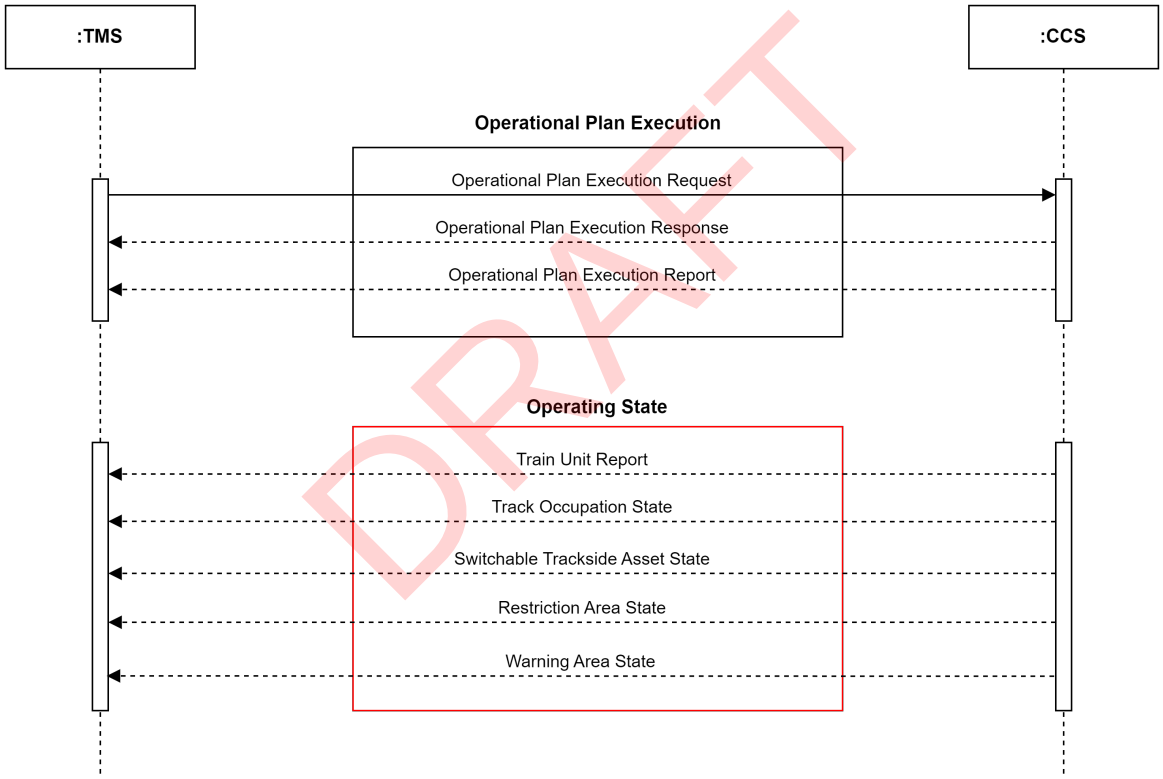
Attributes:

The structure of the message corresponds to: 10.6.1 - Class: EventExecutionReport **[**SPT3TMS-12398 **]**

9 Operating State

Abstract concept “Operating State”:
The Operating State is the logical real-time representation of the actual state of the physical railway system in the Area of Control. **[**SPT3TMS-6675 **]**

The knowledge about the Operating State enables TMS to keep itself current with the operational situation in the Area of Control and to recognise deviations from an Operational Plan during execution. Further, it allows for identifying upcoming or existing conflicts between Operational Plans and developing appropriate countermeasures.
The messages of the Operating State from ATO Trackside and Plan Execution System are embraced into five abstract concepts, as illustrated in Figure 11. **[**SPT3TMS-12256 **]**



[SPT3TMS-6674 **]**

Figure 13 Main abstract concepts of TMS <> CCS with focus on the Operating State

The Operating State provides information of the positions and states of train units, track occupancies, the state of restriction- and warning areas, and the states of the switchable trackside assets, as soon as the information is available. The messages are embraced in five abstract concepts, and generated asynchronously. For every abstract concept, one or more messages exist. They are described in detail in the following sections. While all five abstract

concepts of the Operating State include messages coming from Plan Execution System, the messages of the Train Unit Report are also delivered from ATO Trackside. **[**SPT3TMS-9807 **]**

9.1 Train Unit Report

Abstract concept “Physical Train Unit”:

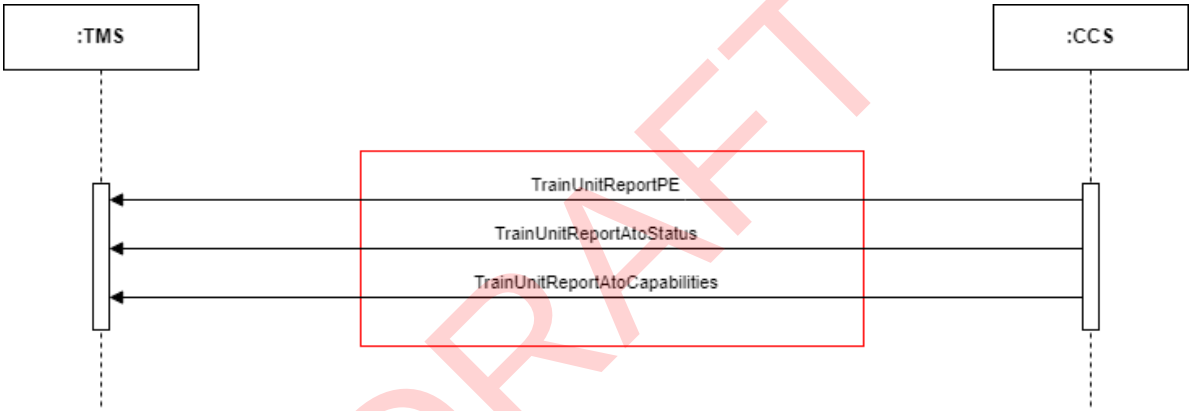
The Physical Train Unit identifies the actual train unit in the railway network. **[**SPT3TMS-6677 **]**

A Physical Train Unit can further be described in terms of Physical Consists and Physical Vehicles if ATO provides corresponding information. **[**SPT3TMS-13961 **]**

Abstract concept “Train Unit Report”:

The Train Unit Report describes the position, properties, and state of any identified Physical Train Unit in the Area of Control. **[**SPT3TMS-13960 **]**

The Train Unit Report consists of three upstream messages as shown in Figure 12: **[**SPT3TMS-12293 **]**



[SPT3TMS-6676 **]**

Figure 14 Upstream messages of Train Unit Report

The messages of a Train Unit Report shall include a reference to the corresponding Operational Plan, if it exists. A Train Unit Report message used without reference to any Operational Plan enables TMS to react unplanned movement, e.g., by issuing a new suitable Operational Plan or by triggering actions for incident prevention.
The three message types of the Train Unit Report are described in the following subsections.

[SPT3TMS-9808 **]**

There are three types of train unit reports:

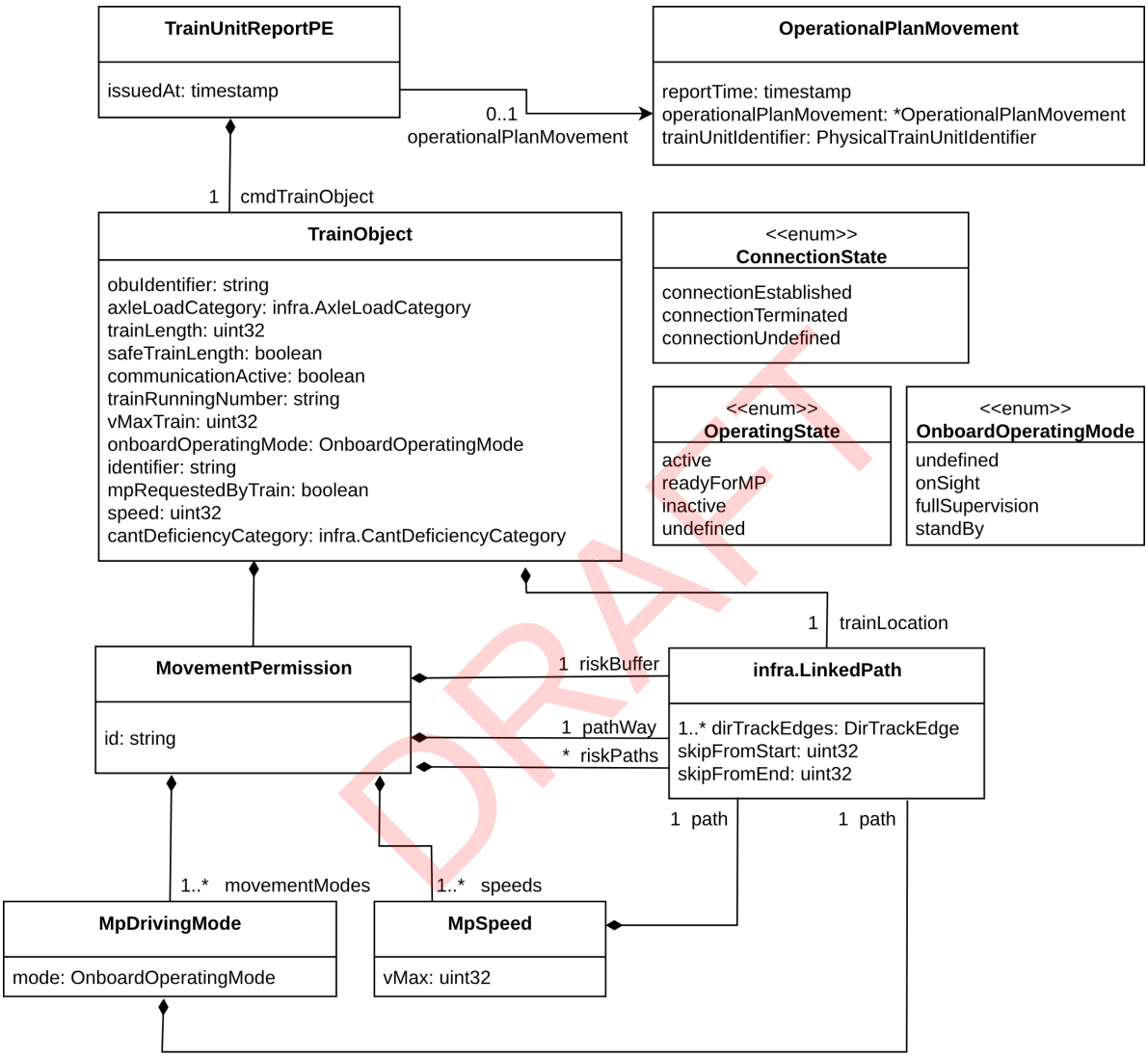
TrainUnitReportPE	TrainUnitReportAtoStatus	TrainUnitReportAtoCapabilities

[SPT2TS-125702 **]**

9.1.1 Message: Train Unit Report PE

Description:

This message describes the ETCS specific information of an identified Physical Train Unit. The message is provided by Plan Execution System. **[**SPT3TMS-9809 **]**



[SPT2TS-126819 **]**

Figure 15 : Class diagram for Train Unit Report PE

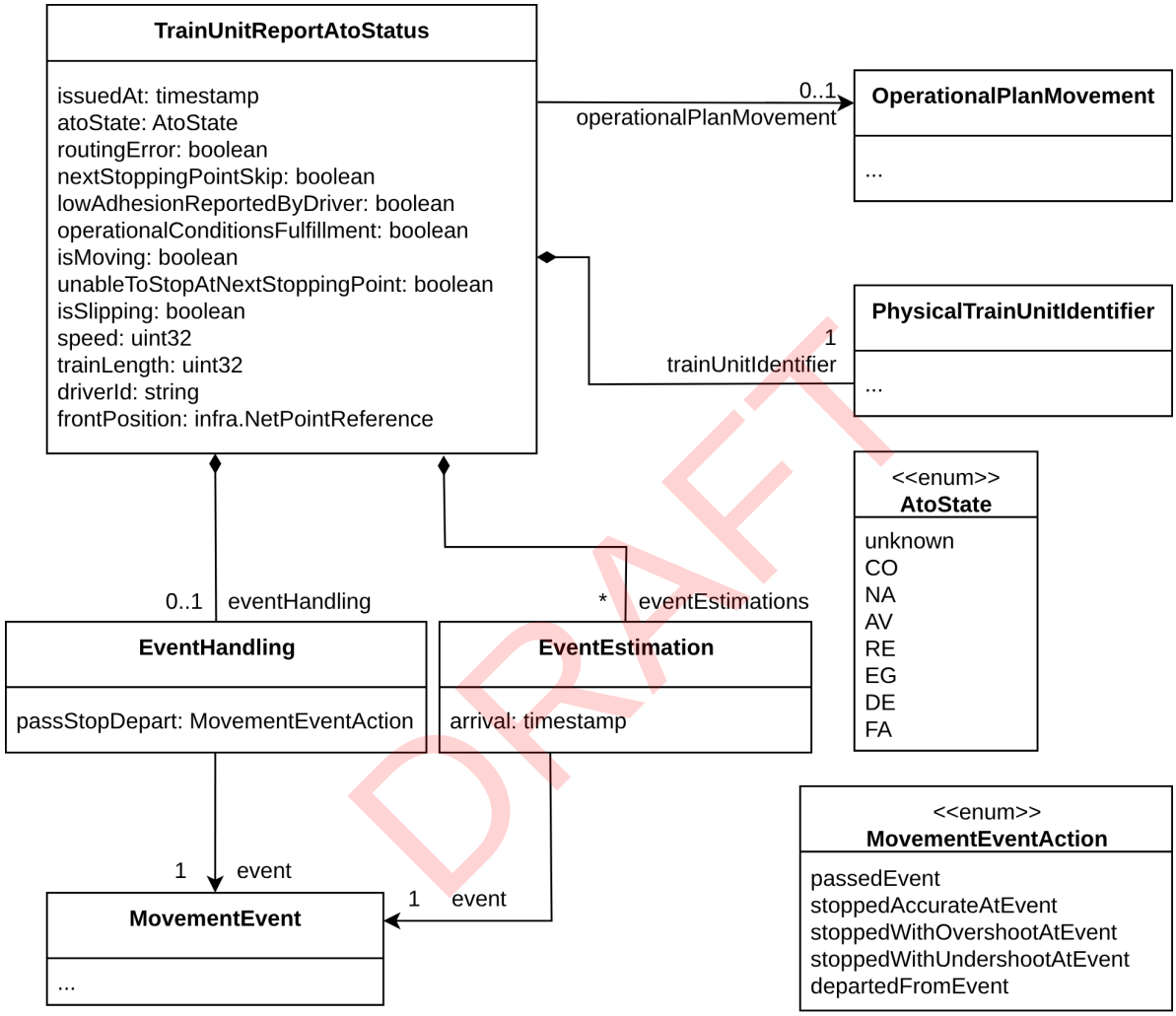
Attributes:

The structure of the message corresponds to: 10.7.1 - Class: TrainUnitReportPE
[SPT3TMS-12407 **]**

9.1.2 Message: Train Unit Report ATO Status

Description:

This message describes an identified Physical Train Unit based on the ATO Status Report (AoE SS-126). The message is provided by ATO Trackside. **[**SPT3TMS-6670 **]**



[SPT2TS-126820 **]**

Figure 16 : Class diagram for Train Unit Report ATO status

Attributes:

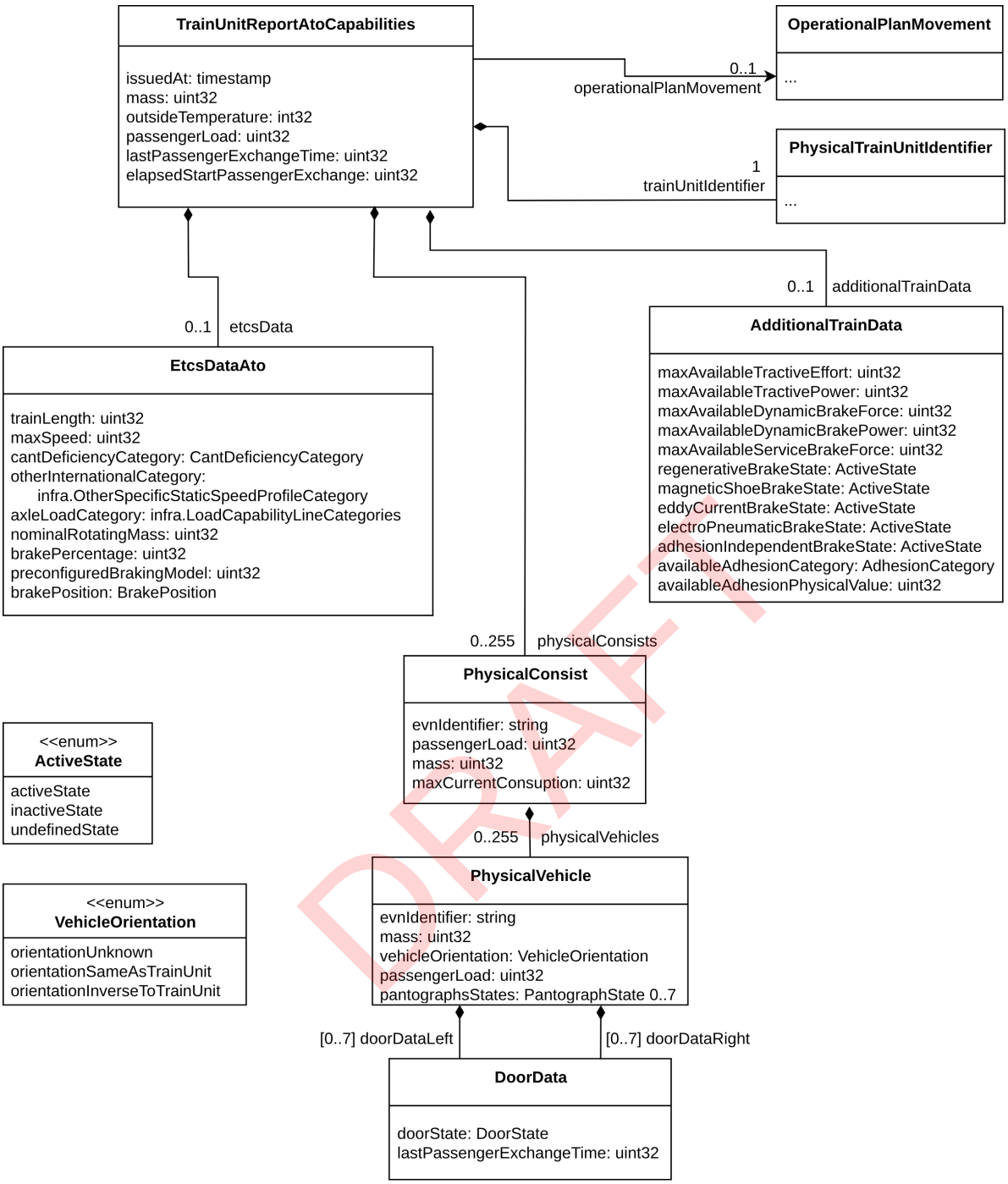
The structure of the message corresponds to: 10.8.1 - Class: TrainUnitReportAtoStatus
[SPT3TMS-15930 **]**

9.1.3 Message: Train Unit Report ATO Capabilities

The Train Capability Report is a proposal for the extension of the existing AoE Subsets. The integration of the Train Capability Report into AoE Subsets and its provision via TMS <> CCS to TMS will be further elaborated in future releases together with the relevant stakeholders.

Description:

This message describes an identified Physical Train Unit based on ATO Train Capability Report (TCR). The message is provided by ATO Trackside. **[**SPT3TMS-6673 **]**



[SPT2TS-125693 **]**

Figure 17 : Class diagram for Train Unit Report ATO Capabilities

Attributes:

The structure of the message corresponds to: 10.9.1 - Class: TrainUnitReportAtoCapabilities
[SPT3TMS-15929 **]**

9.2 Track Occupation State

Abstract concept “Track Occupation State”:

The Track Occupation State describes an track occupation in the Area of Control which is currently associated or not associated to an identified Physical Train Unit. **[**SPT3TMS-6672 **]**

[SPT3TMS-16930 **]**

Figure 18 : Upstream messages for Track Occupation State

The Track Occupation State provides the TMS with information about the creation, update or removal of a track occupation, which cannot be clearly assigned to an identified Physical Train Unit. Messages of Track Occupations shall include a reference to the corresponding Operational Plan if it exists. A Track Occupation message without reference to any Operational Plan enables TMS to react on it, e.g., by issuing a suitable Operational Plan or by triggering actions at Incident Prevention Management. **[**SPT3TMS-9841 **]**

9.2.1 Message: Track Occupation State

Description:

This message reports the occupation of a Track Edge(s) within the Area of Control which cannot be assigned to an identified Physical Train Unit. The message is used for creation, removal and for updates of the same Track Occupation. For each Track Occupation State sent, the Plan Execution System provides the reference to the corresponding Operational Plan, if available. **[**SPT3TMS-6594 **]**

Attributes:

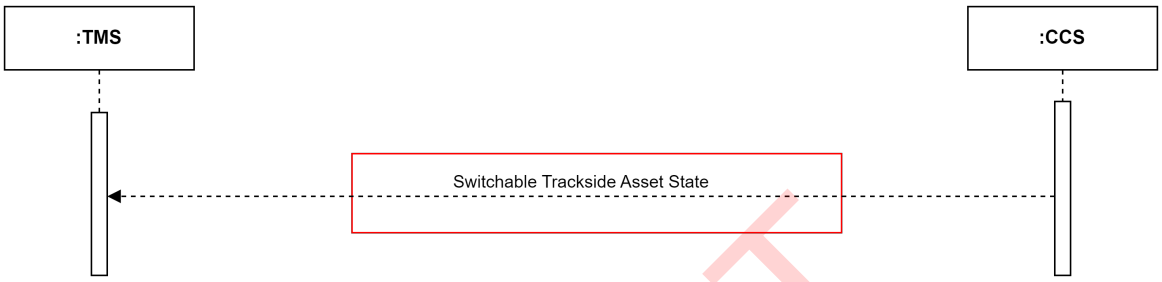
The structure of the message corresponds to: Class: UnresolvedTrackboundObject **[**SPT3TMS-12158 **]**

9.3 Switchable Trackside Asset (Section) State

Abstract concept “Switchable trackside asset section state”

The switchable trackside asset section state describes the state of any switchable trackside asset section in the Area of Control. **[**SPT3TMS-6602 **]**

The switchable trackside asset section state provides the states of a group of switchable trackside assets to TMS. For a point, for example, the position of the point blades is indicated. Each switchable trackside asset section is uniquely identifiable by an ID and belongs to one of predefined Switchable Trackside Asset section Types. **[**SPT3TMS-12429 **]**



[SPT3TMS-16932 **]**

Figure 19 : Upstream message of Switchable Trackside Asset State

9.3.1 Message: Switchable Trackside Asset State

Description:

This message reports an update of the states of a switchable trackside asset section. For each state sent, the Plan Execution System provides the reference to the corresponding Operational Plan, if available. **[**SPT3TMS-9842 **]**

Attributes:

The structure of the message corresponds to: 10.11.1 - Class: SwitchableTracksideAssetState **[**SPT3TMS-14237 **]**

9.4 Restriction Area State

Abstract concept "Restriction Area State:

The Restriction Area State describes the state of an Restriction Area within the Area of Control.
[SPT3TMS-15469]

[SPT3TMS-6681]

Figure 20 Upstream messages of Restriction Area State

The Restriction Area State provides the TMS with information about the status of all Restrictions Areas within the Area of Control. For each Restriction Area State sent, the Plan Execution System provides the reference to the corresponding Operational Plan, if available.
[SPT3TMS-13963]

The message of the Operational Restriction Area is described in the following subsections.
[SPT3TMS-14236]

9.4.1 Message: Restriction Area State

Description:

This message reports the creation, update or removal of a Restriction Area. For each state sent, the Plan Execution System provides the reference to the origin Operational Plan Restriction, if available. The time of the effective Restriction Area creation, update or removal can deviate from the planned creation, update or removal of the corresponding Restriction Area in TMS. This is useful information to identify differences between the planned and the effective start/end of a Restriction Area. **[**SPT3TMS-6680 **]**

Attributes:

The structure of the message corresponds to: 10.12.1 - Class: RestrictionAreaState **[**SPT3TMS-14253 **]**

9.5 Warning Area State

Abstract concept "Warning Area State:

The Warning Area State describes the state of an Warning Area within the Area of Control.
[SPT3TMS-6686 **]**

[SPT3TMS-6612 **]**

Figure 21 Upstream messages of Warning Area State

The Warning Area State provides the TMS with information about the creation, update or removal of a Warning Area within the Area of Control. For each Warning Area State sent, the Plan Execution System provides the reference to the corresponding Operational Plan, if available. **[**SPT3TMS-9838 **]**

9.5.1 Message: Warning Area State

Description:

This message reports the creation, update or removal of a Warning Area. For each state sent, the Plan Execution System provides the reference to the origin Operational Plan Warning Measure, if available. **[**SPT3TMS-6611 **]**

Attributes:

The structure corresponds to: 10.13.1 - Class: WarningAreaState **[**SPT3TMS-14252 **]**

10 Data Model

This chapter describes the defined data model for the implementation of abstract concepts and messages which have been described in chapters 7, 8and 9. The data model uses abstract classes as base classes for other classes. They are instantiated on their own. Abstract classes need to be extended and realised by appropriate classes.
Each class of data model is supplemented by an example. The examples are presented using JSON. However, the format of messages is not predetermined and should be considered for better understanding the data model. **[**SPT3TMS-9830 **]**

10.1 Operational Plan

10.1.1 Abstract Class: OperationalPlan

Description:

This abstract class shall be realized by one of the following classes:

- Operational Plan Movement (see section SPT3TMS-6499 - Operational Plan Movement)
- Operational Plan Restriction (see section SPT3TMS-6519 - Operational Plan Restriction)
- Operational Plan Warning Measure (see section SPT3TMS-6524 - Operational Plan Warning Measure)

[SPT3TMS-9829 **]**

10.2 Operational Plan Movement

10.2.1 Class: OperationalPlanMovement

Description:

This class defines the implementation of an Operational Plan Movement, meaning with this the characterization of a train run with a specific train number. **[**SPT3TMS-6622 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
id	string	1	Unique identifier of Operational Plan for the communication across systems.	Each Operational Plan shall be uniquely identifiable
issuedAt	timestamp	1	Issue date of Operational Plan.	When an Operational Plan will be updated, it must be clearly apparent which is the latest version of this Plan.
configurationDataVersionRef	string	1	Describes the infrastructure version used in the Operational Plan	Verify the validity of the Operational Plan in terms of topology infrastructure.
includesEndOfJourney	boolean	1	If true, the movement events contain the whole remaining path including the end position, otherwise the journey will continue after the last movementEvent of this object.	To give a complete or anyway non-ambiguous description of the train path.
operationalTrainUnits	TrainUnit	1..*	List of Train Units belonging to the Operational Movement.	Characteristics of Train Units are required for implementation and triggering of requests via SCI-CMD.
movementEvents	MovementEvent	1..*	Sequence of Movement Events to be implemented. Each movement event describes an operational activity along the train run.	Characteristics of Movement Events are required for implementation and triggering of requests via SCI-CMD.
movementRestrictions	MovementRestriction	0..*	Sequence of Movement Restrictions to be implemented. Each movement restriction defines an active restriction	Characteristics of Operational Plan Restrictions are required for implementation and

Name:	Type:	Multiplicity:	Description:	Reason:
			which applies to the train along its path.	triggering of requests via SCI-CMD.

[SPT3TMS-11191 **]**

Example: [SPT3TMS-12113 **]**

Class "OperationalPlanMovement"

{
"structs": [
{
"name": "OperationalPlanMovement",
"info": "Defines a train run with a dedicated train number.",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Defines the Identity of the object; used for referencing"},
{"intId": 2, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 3, "name": "configurationDataVersionRef", "dataType": "string", "info": "Defines the configuration data version for which the operational plan is valid"},
{"intId": 4, "name": "includesEndOfJourney", "dataType": "boolean", "info": "Defines if the operational plan movement include the end of journey. if true, the movementEvents contains entire remaining path including the end position, otherwise the journey will continue after the last movementEvent of this object"},
{"intId": 5, "name": "operationalTrainUnits", "composition": "TrainUnit", "multiplicity": "1..*","info": "Defines the set of train units, which are active during some paths of the train movement. The same train unit can be used in not connected parts of the path"},
{"intId": 6, "name": "movementEvents", "composition": "MovementEvent", "multiplicity": "1..*", "ordered": "byIndex", "info": "Defines times/operational activities sequence along the movement"},
{"intId": 7, "name": "movementRestrictions", "composition": "MovementRestriction", "multiplicity": "*", "info": "Defines train specific restrictions, which are active along its path-sections"}
]
}]
}

[SPT2TS-124419 **]**

10.2.2 Class: TrainUnit

Description:

This class describes operational characteristics of an Operational Train Unit. **[**SPT3TMS-6624 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
physicalTrainUnit	PhysicalTrainUnitIdentifier	1	Reference to associated Physical Train Unit	trainRunningNumber/ additionalTrainRunningNumber/tafTapTsiTrainID: Information shall be available in CCS user interface of Workbench (fallback level for TMS)
operationalTrainCategory	OperationalTrainCategory	0..1	Description of the train category.	Information shall be available in CCS user interface of Workbench (fallback level for TMS).
operationalConsists	OperationalConsist	1..*	List of Operational Consists belonging to the same Operational Train Unit.	Characteristics of Operational Consists are required for implementation and triggering of requests via SCI-CMD.
nominalRotatingMass	uint32 (kg)	1	It defines the nominal rotating mass as a percentage of the total train weight. unit is expressed in kg. Possible values are: · Minimal value: 0% · Maximum value: 15% · Resolution: 1% · Special value: 16% = Unknown	For implementation and triggering of requests via SCI-CMD.
regularBrakeWeightPercentage	uint32	1	Definition of the brake weight percentages for the whole Physical Train Unit as a "fixed formation". Value "0" to be used if undefined	For implementation and triggering of requests via SCI-CMD.

Name:	Type:	Multiplicity:	Description:	Reason:
emergency BrakeWeighTPercentage	uint32	1	Definition of the emergency brake weight percentages for the whole Physical Train Unit as a "fixed formation". Value "0" to be used if undefined	For implementation and triggering of requests via SCI-CMD.
brakePosition	BrakePosition	1	Qualifier for brake position	For implementation and triggering of requests via SCI-CMD.
brakeModelType	BrakeModelType	1	Qualifier for gamma/lambda discrimination.	For implementation and triggering of requests via SCI-CMD.
atoGradeOfAutomation	ATOGradeOfAutomation	0..1	Used grade of automation for Physical Train Unit.	For implementation and triggering of requests via SCI-CMD.
scheduledTrainLength	uint32 (m)	1	Train length in meter. May not reflect actual length.	For implementation and triggering of requests via SCI-CMD.

[SPT3TMS-11192 **]**

Example: [SPT3TMS-12115 **]**

Class "TrainUnit"

{"structs": [
{
"name": "TrainUnit",
"info": "Defines rolling stock, which is used to implement part of a train movement.",
"attrs": [
{"intId": 1, "name": "physicalTrainUnit", "composition": "PhysicalTrainUnitIdentifier",
"info": "Defines the physical train unit to be used in Plan Execution user interface of Workbench (fallback level for TMS)"},
{"intId": 2, "name": "operationalTrainCategory", "composition": "OperationalTrainCategory",
"info": "Defines the category type for the train in operational plan. To be used in Plan Execution user interface of Workbench (fallback level for TMS)"},
{"intId": 3, "name": "operationalConsists", "composition": "OperationalConsist", "multiplicity": "1..*",
"ordered": "byIndex", "info": "Defines operational consists building the TrainUnit starting from the train-head."},
{"intId": 4, "name": "nominalRotatingMass", "dataType": "uint32", "unit": "kg", "info": "Defines the nominal rotating mass of the train, special value 0=unknown"},
{"intId": 5, "name": "regularBrakeWeightPercentage", "dataType": "uint32", "unit": "permill", "info": "Defines

the regular brake weight percentage. Use 0 if undefined"},
{"intId": 6, "name": "emergencyBrakeWeightPercentage", "dataType": "uint32", "unit": "permill", "info": "Defines the emergency brake weight percentage. Use zero if undefined"},
{"intId": 7, "name": "brakePosition", "enumType": "BrakePosition", "info": "Defines the active brake position for the train"},
{"intId": 8, "name": "brakeModelType", "enumType": "BrakeModelType", "info": "Defines the applicable brake model type for the train"},
{"intId": 9, "name": "atoGradeAutomation", "enumType": "ATOGradeOfAutomation", "info": "Defines the ato grade of automation for the train"},
{"intId": 10, "name": "scheduledTrainLength", "dataType": "uint32", "unit": "m", "info": "Defines the scheduled train length in meters", "ontology": {"subPropertyOf": "http://data.europa.eu/949/length"}}
]
}]
} **[**SPT2TS-125299 **]**

10.2.2.1 Enum: BrakePosition

Description:

This enum describes possible brake positions. [SPT3TMS-11863]

Attributes:

Name:	Description:
passengerTrainInP	Brake Position P (Passenger) for Passenger Trains
passengerTrainInR	Brake Position R (Rapid) for Passenger Trains
freightTrainInP	Brake Position P (Passenger) for Freight Trains
freightTrainInG	Brake Position G (Goods) for Freight Trains
brakePositionUnknown	Unknown Brake Position

[SPT3TMS-12094]

10.2.2.2 Enum: BrakeModelType

Description:

This enum describes possible brake model types. [SPT3TMS-11867]

Attributes:

Name:	Description:
lambda	Lambda brake model
gamma	Gammer brake model
undefined	Undefined

[SPT3TMS-12093]

10.2.2.3 Enum: ATOGradeOfAutomation

Description:

This enum describes possible grade of automation. [SPT3TMS-11868]

Attributes:

Name:	Description:
GoAUnknown	Unknown grade of automation
GoA1	GoA Level 1
GoA2	GoA Level 2
GoA3	GoA Level 3
GoA4	GoA Level 4

[SPT3TMS-12095 **]**

10.2.3 Class: OperationalTrainCategory

Description:

This class describes the Operational Train Category of a Train Unit. [SPT3TMS-6626]

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
type	TrainCategoryType	0..1	Corresponds to Traffic Types as defined in TAF/TAP TSI. The aligned values may be translated with an HMI to nationally established, human-readable short strings, giving the object a name (e.g. Regionalbahn, Regionaltog, Snabbtåg, Intercity, Frecciarossa, Railjet, Eurostar).	Information shall be available in Plan Execution System user interface of Workbench (fallback level for TMS)
description	string	0..1	Human-readable, more detailed description as addition to the name. Not intended for machine interpretation.	Information shall be available in Plan Execution System user interface of Workbench (fallback level for TMS)
trainUsage	TrainUsageType	0..1	Purpose of the train journey for service, representation, and information applications.	Information shall be available in Plan Execution System user interface of Workbench (fallback level for TMS). It is not intended to pass this information to other sub-systems.

[SPT3TMS-11190]

Example: [SPT3TMS-12117]

Class OperationalTrainCategory

{"structs": [
{
"name": "OperationalTrainCategory",
"info": "Defines usage category for an OperationalTrain",
"attrs": [
{"intId": 1, "name": "trainCategoryType", "enumType": "TrainCategoryType"},
{"intId": 2, "name": "description", "dataType": "string", "multiplicity": "0..1", "info": "Provides a description of the operational train category for user interface display, used as a fallback for manual route setting."},
{"intId": 3, "name": "trainUsage", "enumType": "TrainUsageType", "info": "Defines the train usage type for UI-purposes, used as a fallback for manual route setting."}
]

}]
} **[**SPT2TS-125301 **]**

10.2.3.1 Enum: TrainCategoryType

Description:

This enum describes possible train category types. [SPT3TMS-11870]

Attributes:

Name:	Description:
trainCategoryUnknown	Train Category unknown
interregional	Inter-Regional
regional	Regional
subUrban	Sub-Urban
nightTrain	Night train
motorRail	Motor rail
mountainTrain	Mountain train
historicTrain	historical train
coachGroup	Coach group
tram	Tram
underground	Underground train
highspeedTrain	Highspeed train
intercity	Intercity train

[SPT3TMS-12097]

10.2.3.2 Enum: TrainUsageType

Description:

This enum describes possible train usage types. [SPT3TMS-11872]

Attributes:

Name:	Description:
trainUsageUndefined	Not defined
commercialPassengerTrain	Passenger usage

Name:	Description:
commercialCargoTrain	Cargo usage
locomotiveRunningLight	Locomotive running light
lightRunning	Light Running
notInService	Not in Service
engineeringTrain	Engineering train
breakdownTrain	Break down train
mixedTrain	Mixed train
specialTrain	Special train
otherTrain	Other train

[SPT3TMS-12096 **]**

10.2.4 Class: OperationalConsist

Description:

The class references an Operational Consist and defines its position in the Train Unit. **[**SPT3TMS-6627 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
supportedOnboardEquipment	SupportedOnboardEquipment	1	Reference to supported onboard Equipment	-

[SPT3TMS-11187 **]**

Example: [SPT3TMS-12116 **]**

Class "OperationalConsist"

{"structs": [
{
"name": "OperationalConsist",
"info": "Defines a rolling-stock consist unit identified by the installed onboard equipment.",
"attrs": [
{"intId": 1, "name": "supportedOnBoardEquipment", "composition": "SupportedOnBoardEquipment", "info": "Defines the list of supported onboard equipment for an operational consist"}
]
}]
} **[**SPT2TS-125305 **]**

10.2.5 Class: PhysicalTrainUnitIdentifier

Description:

This class describes the identification parameters of a Physical Train Unit. **[**SPT3TMS-6635 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
trainRunningNumber	string	0..1	Operational identification of the Physical Train Unit. This identifier is what in SS026 is indicated as NID_OPERATIONAL (please note that it has not necessarily to be unique)	Information shall be available in CCS user interface of Workbench (fallback level for TMS)
additionalTrainRunningNumber	string	0..1	Distinction between Operational Train Units with the same trainRunningNumber. This permits to uniquely identify a train distinguishing among Train Units with the same trainRunningNumber. This may occur if a trainRunningNumber is reused for different spare trains. As an example, this can be used to indicate Ante ("A"), Bis ("B"), Ter ("C") trains added to a first train to handle specific crowded situations.	Information shall be available in CCS user interface of Workbench (fallback level for TMS)
tafTapTsiTrainID	string	0..1	Unique TrainID, used for the explicit identification of a Physical Train Unit, especially for international train runs. The number will be implemented during the implementation of the TAF/TAP TSI.	Information shall be available in CCSuser interface of Workbench (fallback level for TMS)
leadingOnBoardUnitId	string	0..1	Unique onboard ID (NID_ENGINE).	Information shall be available to uniquely identify the relevant Physical Train Unit

[SPT3TMS-11788 **]**

Example: [SPT3TMS-12123 **]**

Class "PhysicalTrainUnitIdentifier"

{"structs": [
{
"name": "PhysicalTrainUnitIdentifier",
"info": "Defines a set of possible identifiers for a physical train (ETCS-trainRunningNumber, OBU-ID etc).",
"attrs": [
{"intId": 1, "name": "trainRunningNumber", "dataType": "string", "multiplicity": "0..1", "info": "Defines the Train Running Number for ETCS same as SS026.NID_OPERATIONAL. To be usable for ETCS must contain an integer [0..99999999]. Mapping to uint32 by filling with F: '1233' -> 0x1233FFFF"},
{"intId": 2, "name": "additionalTrainRunningNumber", "dataType": "string", "multiplicity": "0..1", "info": "Defines the distinction between Operational Train Units with the same trainRunningNumber. This may

occur if a trainRunningNumber is reused for different spare trains."},
{"intId": 3, "name": "tafTapTsiTrainID", "dataType": "string", "multiplicity": "0..1", "info": "refers to taf tap tsi train id"},
{"intId": 4, "name": "leadingOnBoardUnitId", "dataType": "string", "multiplicity": "0..1", "info": "Defines NID_ENGINE. To be usable with ETCS must contain an integer [0..16.777.215]"}
]
}]
} **[**SPT2TS-125308 **]**

10.2.6 Class: SupportedOnBoardEquipment

Description:

This class describes the technical on-board equipment supported by an Operational Consist such as the installed train protection system equipment or ATO-related equipment. It is provided to the CCS in an Operational Plan Movement. **[**SPT3TMS-6628 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
trainProtectionType	TrainProtectionType	1	This specifies the type of the train protection system using an enumeration to have unique names.	For implementation and triggering of requests via SCI-CMD (Movement Permission)
etcsSystemVersions	string	1..*	Delivers the list of System Versions supported by the Operational Consist. String is formatted according ot the following rule: ""regex": "\\d+.\\d+""	For implementation and triggering of requests via SCI-CMD (Movement Permission)
onboardExtensions	OnboardExtension	0..*	Defines an extension point for a non-TMS <> CCS class	Extension point for e.g. for further used on-board equipment

[SPT3TMS-11194 **]**

Example: [SPT3TMS-12122 **]**

Class SupportedOnBoardEquipment

{"structs": [
{
"name": "SupportedOnBoardEquipment",
"info": "Defines Onboard equipment installed and active on a train",
"attrs": [
{"intId": 1, "name": "trainProtectionType", "enumType": "TrainProtectionType", "info": "Defines the train protection type supported by the on board equipment"},
{"intId": 2, "name": "etcsSystemVersions", "dataType": "string", "multiplicity": "1..*", "regex": "\\d+\\.\\d+", "info": "Defines the supported etcs system versions"},
{"intId": 3, "name": "onboardExtensions", "composition": "OnBoardExtension", "multiplicity": "*"}
]
}]
} **[**SPT2TS-125306 **]**

10.2.6.1 Enum: TrainProtectionType

Description:

This enum describes possible train protection equipment. [SPT3TMS-11937]

Attributes:

Name:	Description:
etcs	European Train Control System

[SPT3TMS-12099]

10.2.7 Class: OnBoardExtension

Description:

This class describes potential extensions to the Train Unit. [SPT3TMS-11995]

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
contentType	string	1	Class Extension	-
content	bytes	1	Content of class Extension	-

[SPT3TMS-11996]

Example: [SPT3TMS-12125]

Class "OnBoardExtension"

{"structs": [
{
"name": "OnBoardExtension",
"info": "A container class for a non-standard key-value-property. Messages need sometimes IM-specific information.",
"attrs": [
{"intId": 1, "name": "contentType", "dataType": "string", "info": "Defines the key-attribute in a key-value-pair. Used for annotation of IM-specific onboard-equipment"},
{"intId": 2, "name": "content", "dataType": "bytes", "info": "Defines the value-attribute in a key-value-pair. Used for annotation of IM-specific onboard-equipment"}
]
}]
} [SPT2TS-125307]

10.2.8 Class: MovementEvent

Description:

This class describes the number, position and characteristics of a Movement Event.
[SPT3TMS-6630 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
id	string	1	Universally unique identifier used for identification of the object across systems.	Unique identifier for the communication across systems.
plannedType	MovementEventType	1	Describes the planned action of the Physical Train Unit at this Movement Event.	For implementation and triggering of requests via SCI-CMD.
scheduledType	MovementEventType	1	Describes the scheduled action of the Physical Train Unit at this Movement Event.	For implementation and triggering of requests via SCI-CMD.
alignment	AlignmentType	1	Describes the alignment of the arriving train in relation to the place on a track referenced by Movement Event.	For implementation and triggering of requests via SCI-CMD.
position	infra.NetPointReference	1	Describes the position referenced by the Movement Event.	For implementation and triggering of requests via SCI-CMD.
pathToNextEvent	infra.LinearElement	0..*		-

Name:	Type:	Multiplicity:	Description:	Reason:
			Sequence of next track paths events. The event.position is on the [0]-element and nextEvent.position is on the [last]-element.	
stopDescription	StopDescription	0..1	Describes the characteristics and activities to be carried out at planned stop of a Physical Train Unit	For implementation and triggering of requests via SCI-CMD. For implementation and triggering of requests via AoE SS-126.
scheduledArrival	timestamp	1	Describes the latest point of arrival. Equal to scheduledDeparture for a passing train, see latestArrival in SS126. In the first station it is time of 'start of mission'	-
scheduledArrivalWindow	uint32 (s)	1	Describes how much time in seconds before the latestArrival the Movement Event may be reached. ScheduledEaliestArrival = scheduledArrival - scheduledArrivalWindow. For passing trains scheduledEarliestDeparture = scheduledEarliestArrival, see arrivalWindow in SS126. In the first station it is a time window for 'start of mission'.	-
startsAfterEvents	OperationalEventRef	0..*	List of dependencies between this Event and any other Event of the same or another Operational Plan.	For implementation and triggering of requests via SCI-CMD.

[SPT3TMS-12098 **]**

10.2.9 Class: OperationalEventRef

Description:

This union describes the dependency between the origin Operational Event and any other Operational Event of the same or another Operational Plan.

Please note: Only unidirectional dependencies are expected; a link from the referenced event back to the original one is not foreseen. **[**SPT3TMS-16517 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
movement Event	Movement Event	1	refers a movement event including the movement plan	To force a dependency (order) between this movement and another movement event
restrictionE vent	Restriction Event	1	Refers a restriction event including restriction plan	To force a dependency (order) between this movement and a restriction event
warningMe asureEvent	WarningM easureEve nt	1	Refers a warning measure event including the warning plan	To force a dependency (order) between this movement and a warning measure event

[SPT3TMS-16519 **]**

Example: [SPT3TMS-12124 **]**

Class "Movement Event"

{
"structs": [
{
"name": "MovementEvent",
"info": "Defines an operational activity on a specific point in space and time for a train run.",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Defines the Identity of the object; used for referencing"},
{"intId": 2, "name": "plannedType", "enumType": "MovementEventType", "info": "Defines if stopping or passing was planned"},
{"intId": 3, "name": "scheduledType", "enumType": "MovementEventType", "info": "Defines if stopping or passing is scheduled"},
{"intId": 4, "name": "alignment", "enumType": "AlignmentType", "info": "Defines the train-alignment at position-attribute (head, center, end)"},
{"intId": 5, "name": "position", "composition": "infra.NetPointReference", "info": "Defines position on a linearElement for the movement event"},
{"intId": 6, "name": "pathToNextEvent", "reference": "infra.LinearElement", "multiplicity": "0..*", "ordered": "byIndex", "info": "Defines the path-part **after** the LinearElement of the position-value and **before** the linear element of the nextEvent (excluding). The path must be reconstructable without graph-search algorithms. The path remains empty, if 1) same linearElement in both events, 2) no driving activity in between e.g. in Joining, Splitting, turnAround, trainReplacement."},
{"intId": 7, "name": "stopDescription", "composition": "StopDescription", "multiplicity": "0..1", "info": "Defines the stop description in case of a stop of the train"},
{"intId": 8, "name": "scheduledArrival", "dataType": "timestamp", "info": "Defines the scheduled arrival time. equal to scheduledDeparture for a passing train, see latestArrival inSS126. In the first station it is time of 'start of mission', in UTC"},

{"intId": 9, "name": "scheduledArrivalWindow", "dataType": "uint32", "unit": "s", "info": "Defines the scheduled arrival window. scheduledEarliestArrival = scheduledArrival - scheduledArrivalWindow. For passing trains scheduledEarliestDeparture = scheduledEarliestArrival, see arrivalWindow in SS126. In the first station it is a time window for 'start of mission'."},
{"intId": 10, "name": "startsAfterEvents", "composition": "OperationalEventRef", "multiplicity": "0..*", "info": "Defines the set of operational events, which must be finished before the current event starts."}
]
},
{
"name": "OperationalEventRef",
"info": "Defines a reference to one of possible OperationalEvents (as a unit)",
"union": true,
"attrs": [
{"intId": 1, "name": "movementEvent", "reference": "MovementEvent", "info": "refers a movement event including the movement plan"},
{"intId": 2, "name": "restrictionEvent", "reference": "RestrictionEvent", "info": "Refers a restriction event including restriction plan"},
{"intId": 3, "name": "warningMeasureEvent", "reference": "WarningMeasureEvent", "info": "Refers a warnning measure event including the warning plan"}
]
}
]
} **[**SPT2TS-124423 **]**

10.2.9.1 Enum: MovementEventType

Description:

This enum describes the possible movement event types. **[**SPT3TMS-12062 **]**

Attributes:

Name:	Description:
pass	the planned passage of a Physical Train Unit. The Passage Event shall specify at least the position on the track and the time range (upper and lower bound) for the planned passage.
stop	the planned stop of a Physical Train Unit as well as all relevant planned actions to be carried out at planned stop of a Physical Train Unit.

[SPT3TMS-12101 **]**

10.2.9.2 Enum: AlignmentType

Description:

This class describes the sequence number, position and characteristics of a Movement Event. **[**SPT3TMS-12064 **]**

Attributes:

Name:	Description:
head	The head of the train (front of first vehicle) shall be located at the position specified by the Movement Event.
center	The center of the train (midpoint concerning the train length) shall be located at the position specified by the Movement Event.
rear	The end of the train (back of last vehicle) shall be located at the position specified by the Movement Event.

[SPT3TMS-12100 **]**

10.2.10 Class: StopDescription

Description:

This class describes the characteristics and activities to be carried out at a planned stop of a Physical Train Unit. **[**SPT3TMS-6633 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
trainUnitActivities	TrainUnitActivity	*	List of stop activities. They describe activities to be carried out at planned stop of a Physical Train Unit.	For implementation and triggering of requests via AoE SS-126.
doorActivity	DoorActivity	0..1	This indicates a specific door activity at the stop.	For implementation and triggering of requests via AoE SS-126.
relaxedCoupler	boolean	1	The attribute was introduced for locomotive-hauled trains. Normally, the train brake is used to stop these. This means that the pressure in the main brake pipe is lowered, which results in all coupled wagons braking as well and thus the couplings are tensioned when the train is at a standstill. This is not always desired, e.g. if the wagons are to be uncoupled at the stop. However, if the variable Q_Relaxed_Coupler (k,l) is set, ATO will brake below a certain speed with a brake that only acts on the locomotive. This then results in the couplers being relaxed at standstill. Possible values are: • TRUE: request for coupler relaxation • FALSE: no request for coupler relaxation	For implementation and triggering of requests via AoE SS-126.

Name:	Type:	Multiplicity:	Description:	Reason:
holdTrain	boolean	1	If holdTrain = true for a stop, ATO will not depart from this stop until it receives a new JP with holdTrain = false. ATO will also display holdTrain value to the driver. If holdTrain = true for a stop, no departure time is defined in the JP. With "End of Journey", no trainHold can be defined in the JP. Possible values are: • TRUE: hold train • FALSE: do not hold train	For implementation and triggering of requests via AoE SS-126.
scheduledDeparture	timestamp	1	Indicates the departure time. In the last station it is 'end of mission'	-
scheduledMinDwellTime	uint32 (s)	0..1	Minimum dwell time in seconds	For implementation and triggering of requests via AoE SS-126.
additionalEventTimes	AdditionalEventTime	0..5	Indicates additional event timings.	Additional times for C-DAS. They will be ignored by ATO (not part of SS126).

[SPT3TMS-11784]

10.2.11 Class: AdditionalEventTime

Description:

This class describes additional event times. [SPT3TMS-14057]

Attributes:

Name:	Type:	Multiplicity:	Description:
timeValue	timestamp	1	Indicates the timestamp value in combination with the time type.
timeType	TimeType	1	A specific time type.

[SPT3TMS-14059]

10.2.11.1 Enum: TimeType

Description:

This enum describes the possible time types. **[**SPT3TMS-14060 **]**

Attributes

Name:	Description:
plannedArrival	Planned Arrival
plannedDeparture	Planned Departure
plannedMinDwellTime	Planned Min Dwell Time
publishedArrival	Published Arrival
publishedDeparture	Published Departure

[SPT3TMS-14062 **]**

Example: [SPT3TMS-12119 **]**

Class StopDescription

{
"structs": [
{
"name": "StopDescription",
"info": "Defines activities to be implemented by the train when it stops.",
"attrs": [
{"intId": 1, "name": "trainUnitActivities", "composition": "TrainUnitActivity", "multiplicity": "*", "ordered": "byIndex", "info": "Defines the sequence of changes of TrainUnit during the stop"},
{"intId": 2, "name": "doorActivity", "composition": "DoorActivity", "multiplicity": "0..1", "info": "Defines the door activity. If the attribute is not provided, the doors remain closed"},
{"intId": 3, "name": "relaxedCoupler", "dataType": "boolean", "info": "Defines the state of coupler. true - request for coupler relaxation, false -not, see Q_Relaxed_Coupler in SS126"},
{"intId": 4, "name": "holdTrain", "dataType": "boolean", "info": "Defines, if the train shall be hold until the next update. See JP in SS126"},
{"intId": 5, "name": "scheduledDeparture", "dataType": "timestamp", "info": "Defines the scheduled departure time. see departure in SS126. In the last station it is 'end of mission', in UTC."},
{"intId": 6, "name": "scheduledMinDwellTime", "dataType": "uint32", "unit": "s", "info": "Defines if the train has to wait after actual arrival this number of seconds before departure. ActualEarliestDeparture = max(departure, arrival + minDwellTime) will be calculated by ATO-OB, see minDwellTime in SS126"},
{"intId": 7, "name": "additionalEventTimes", "composition": "AdditionalEventTime", "multiplicity": "0..5"}
]
},
{
"name": "AdditionalEventTime",
"info": "Defines times not required for ATO, but useful for CDAS (e.g. published arrival, plannedArrival etc.)",

"attrs": [
{"intId": 1, "name": "timeValue", "dataType": "timestamp", "info": "Defines the time value in UTC"},
{"intId": 2, "name": "timeType", "enumType": "TimeType", "info": "Defines the time type for a stop description"}
]
}],
"enums": [
{
"name": "TimeType",
"enumLiterals": [
{"intId": 0, "name": "plannedArrival"},
{"intId": 1, "name": "plannedDeparture"},
{"intId": 2, "name": "plannedMinDwellTime"},
{"intId": 3, "name": "publishedArrival"},
{"intId": 4, "name": "publishedDeparture"}
]
}
]
} **[**SPT2TS-125298 **]**

10.2.12 Class: DoorActivity

Description:

This class describes the door activity.

If not defined, the doors remain closed, once specified the door may be opened.
[SPT3TMS-13930 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
openingDoorSide	DoorSide	1	noneSide if the doors are kept closed
centralisedOpening	boolean	1	• true opening by passengers/driver • false - centralised
automaticClosing	boolean	1	• true by ATO, • false - without ATO

[SPT3TMS-13932 **]**

Example: [SPT3TMS-13933 **]**

Class DoorActivity

{
"structs": [
{
"name": "DoorActivity",
"info": "Defines which doors shall be open and how they shall be open.",
"attrs": [
{"intId": 1, "name": "openingDoorSide", "enumType": "DoorSide", "info": "Defines the door opening side. noneSide if the doors are kept closed. noneSide if the doors are kept closed"},
{"intId": 2, "name": "centralisedOpening", "dataType": "boolean", "info": "Defines if the doors can be centrally opened. true - centralised opening, false - opening by passengers/driver"},
{"intId": 3, "name": "automaticClosing", "dataType": "boolean", "info": "Defines if automatic closing of doors is performed by ATO or not. True by ATO, false - without ATO"}
]
}
]
}

[SPT2TS-126732 **]**

10.2.12.1 Enum: DoorSide

Description:

This enum describes the possible door sides in running direction. **[**SPT3TMS-12050 **]**

Attributes

Name:	Description:
doorSideLeft	Doors on the left side.
doorSideRight	Doors on the rights side.
doorSideBoth	Doors on both sides.
doorNoneSide	No doors, independent of the side.
doorSideUnknown	Doors side unknown

[SPT3TMS-12107]

10.2.12.2 Enum: DoorState

Description:

This enum describes the possible door states. [SPT3TMS-12603]

Attributes

Name:	Description:
doorStateUnknown	Door state unknown
doorStateBlocked	Door state blocked
doorStateAvailable	Door state available

[SPT3TMS-12605]

10.2.13 Class: TrainUnitActivity

Description:

This class describes an activity carried out at a stop. **[**SPT3TMS-6636 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
trainUnitActivityType	TrainUnitActivityType	1	Describes the activity carried out at this planned stop.	For implementation and triggering of requests via SCI-CMD. “STABLE”: ATO could use the information for shutting down the PTU. PES could request to release a too long MP.
actualTrainUnitIndex	uint32	1	Reference to associated Operational Segment which is valid before the Movement Event was executed	For implementation and triggering of requests via SCI-CMD.
targetTrainUnitIndex	uint32	1	Reference to the associated Operational Segment which will be valid after the Movement Event was executed	For implementation and triggering of requests via SCI-CMD.

[SPT3TMS-11786 **]**

Example: [SPT3TMS-12121 **]**

Class TrainUnitActivity

{
"structs": [
{
"name": "TrainUnitActivity",
"info": "Defines, what happens with the rolling stock at this location.",
"attrs": [
{"intId": 1, "name": "trainUnitActivityType", "enumType": "TrainUnitActivityType", "info": "Defines the train unit activity type"},
{"intId": 2, "name": "actualTrainUnitIndex", "dataType": "uint32", "info": "Defines the index of TrainUnit in OperationalMovement. operationalTrainUnits before the activity. Index starts with 0"},
{"intId": 3, "name": "targetTrainUnitIndex", "dataType": "uint32", "info": "Defines the index of TrainUnit in OperationalMovement. operationalTrainUnits after the activity. Index starts with 0"}
]
}]
} **[**SPT2TS-125313 **]**

10.2.13.1 Enum: TrainUnitActivityType

Description:

This enum describes possible train unit activity types.

Attributes

Name:	Description:
joinActivity	Couple vehicles / train parts - intended for self-propelling train parts. Please consider relation to formations (as far as used)
splitActivity	Uncouple vehicles / train parts - intended for self-propelling train parts. Please consider relation to formations (as far as used)
collectActivity	Couple vehicles / parts
dropActivity	Uncouple vehicles / train parts
turnAroundActivity	Stop to change driving direction of a train. After TurnAround a new MovementEvent with changed train position will be inserted into the Movement
meetActivity	Allows to enter two trains in one track section (TVPS) similar to a join activity without coupling.
trainReplacementActivity	Manages the case when a trainUnit is replaced by another one on another netElement. The new position is provided in the next movementEvent. The pathToNextEvent in the current event remains empty.
otherActivity	When the change must be calculated by comparing TrainUnits before and after the activity.

[SPT3TMS-12110]

10.2.14 Class: MovementRestriction

Description:

Usecase for the class is to define non-safety-relevant restrictions based on speed, adhesion to influence driving-profile more precisely than with timing-points only. **[**SPT3TMS-12054 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
restrictedPath	infra.NetLinearReference	1	Start to end point of movement restrictions
specificRestriction	SpecificMovementRestriction	1	Specific movement restriction

[SPT3TMS-12058 **]**

Example: [SPT3TMS-12120 **]**

Class "MovementRestriction"

{"structs": [
{
"name": "MovementRestriction",
"info": "Defines train specific restriction along its path.",
"attrs": [
{"intId": 1, "name": "restrictedPath", "composition": "infra.NetLinearReference", "info": "Defines the restricted path associated to a restriction of an operational plan movement"},
{"intId": 2, "name": "specificRestriction", "composition": "SpecificMovementRestriction", "info": "Defines the specific restriction on the restriction path associated to an operational plan movement"}
]
}]
} **[**SPT2TS-125302 **]**

10.2.15 Class: SpecificMovementRestriction

Description:

This class describes an specific movement restriction **[**SPT3TMS-12056 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
speedRestriction	SpeedRestriction	0..1	Max speed inside the restriction	-
adhesionCategory	AdhesionCategory	0..1	Adhesion category	For implementation and triggering of requests via AoE SS-126
maxCurrent (A)	uint32 in Amps	1	Electric current limitation, s. item 0.33 in SS126	-
atoInhibition	boolean	1	Train section is not contantly monitored, or stimulate driver's attention, or approaching overcrouded station	-
dasInhibition	boolean	1	Driver advisory system not available in this section	-
excludedSwitchableTracksideAssetsForFlankProtection	tp.SwitchableTracksideAsset	0..*	List of Switchable trackside assets to be excluded from flank protection calculation.	-

[SPT3TMS-12059 **]**

Example: [SPT3TMS-12133 **]**

Class SpecificMovementRestriction

{"structs": [
{
"name": "SpecificMovementRestriction",
"info": "Defines single aspect of the train specific restriction",
"union": true,
"attrs": [
{"intId": 1, "name": "speedRestriction", "composition": "SpeedRestriction"},
{"intId": 2, "name": "adhesionCategory", "enumType": "AdhesionCategory", "info": "Defines the different adhesion categories applicable for the train as per track conditions"},
{"intId": 3, "name": "maxCurrent", "dataType": "uint32", "unit": "A", "info": "Defines max current value, s. item 0.33 in SS126"},
{"intId": 4, "name": "atoInhibition", "dataType": "boolean", "info": "train section is not contantly monitored, or stimulate driver's attention, or approaching overcrouded station"},
{"intId": 5, "name": "dasInhibition", "dataType": "boolean", "info": "Defines the Driver advisory system inhibition zone"},
{"intId": 6, "name": "excludedSwitchableTracksideAssetsForFlankProtection", "reference": "tp.SwitchableTracksideAsset", "multiplicity": "0..*", "info": "Defines the list of SwitchableTracksideAssets to be excluded from flank protection calculation."}
]
}
]
[SPT2TS-125303 **]**

10.2.15.1 Enum: AdhesionCategory

Description:

This enum describes possible adhesion categories. **[**SPT3TMS-12060 **]**

Attributes

Name:	Description
dryRailHigh	Conditions where 100% of the brake force of the vehicle can be applied with no axle sliding of more than 2% (adhesion level typically above 0.15µ)
dryRailMedium	Conditions where the wheel/rail adhesion is in the range 0.15 – 0.10 (Damp rails with some contamination)
dryRailLow	Conditions where the wheel/rail adhesion is in the range 0.10 – 0.08 (Typical autumn mornings due to dew/dampness often combined with light overnight rust)
lowAdhesion	Conditions where the wheel/rail adhesion is in the range 0.08 – 0.05
veryLowAdhesion	Conditions where the wheel/rail adhesion is in the range 0.05-0.03
extremelyLowAdhesion	Conditions where the wheel/rail adhesion is below 0.03
unknownAdhesion	Unknown adhesion

[SPT3TMS-12109 **]**

10.3 Operational Plan Restriction

10.3.1 Class: Restriction

Description:

This class defines the implementation of one or more Operational Plan Restrictions. It relates to one or more Restriction Areas and it contains references to the Restriction Events of these Restriction Areas. **[**SPT3TMS-6637 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
id	string	1	Universally unique identifier used for identification of the object across systems.	Unique identifier for the communication across systems.
issuedAt	timestamp	1	Issued timestamp	-
configurationDataVersionRef	String	1	Reference to domain data version.	-
restrictionEvents	RestrictionEvent	1..*	Sequence of Restriction Events to be implemented	Characteristics of Restriction Events (start and finish) are required for implementation and triggering of requests via SCI-CMD.

[SPT3TMS-11789 **]**

Example: [SPT3TMS-12135 **]**

Class "Restriction"

{
"structs": [
{
"name": "OperationalPlanRestriction",
"info": "Defines a temporary infrastructure restriction",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Defines the Identity of the object; used for referencing"},
{"intId": 2, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 3, "name": "configurationDataVersionRef", "dataType": "string", "info": "Defines the configuration data version for which the operational plan is valid"},
{"intId": 5, "name": "restrictionEvents", "composition": "RestrictionEvent", "multiplicity": "1..*", "info": "Defines a set of restriction events assigned the the operational plan restriction"}
]
}]
}

[SPT2TS-124447 **]**

10.3.2 Class: RestrictionEvent

Description:

This class describes the characteristics of a Restriction Event. A restriction event is a self-contained element of a temporary infrastructure restriction. A wide temporary infrastructure restriction can be made of several restriction events which can be activated/deactivated at different times. **[**SPT3TMS-14063 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	Universally unique identifier used for identification of the object across systems.
creationTime	timestamp	1	Creation time
removalTime	timestamp	1	Removal time
restrictionArea	RestrictionArea	1	List of Restriction Areas to be implemented
startsAfterEvents	OperationalEventRef	0..*	List of dependencies between this Operational Event and any other Operational Event of the same or another Operational Plan. The event can start only after the other events here referred are finished

[SPT3TMS-14065 **]**

Example: [SPT3TMS-14066 **]**

Class RestrictionEvent

{
"structs": [
{
"name": "RestrictionEvent",
"info": "Defines a part of temporary infrastructure restriction, especially if one Restriction plan contains several areas to be activate to different times.",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Defines the Identity of the object; used for referencing"},
{"intId": 2, "name": "creationTime", "dataType": "timestamp", "info": "Defines the time when the restrictionArea shall be created=implemented=activated in interlocking, in UTC"},
{"intId": 3, "name": "removalTime", "dataType": "timestamp", "info": "Defines the removal time in UTC. E.g. removal 10:04:20 means, that restriction is active until 10:04:19.999999. The relation is creationTime <= active < removalTime"},
{"intId": 4, "name": "restrictionArea", "composition": "RestrictionArea", "info": "Defines the restriction area

associated to a restriction event"},
{"intId": 5, "name": "startsAfterEvents", "composition": "OperationalEventRef", "multiplicity": "0..*", "info": "Defines the set of operational events, which must be finished before the current event starts."}
]
}
]
}

[SPT2TS-126703 **]**

10.3.3 Class: RestrictionArea

Description:

This class describes the characteristics of a Restriction Area. [SPT3TMS-6639]

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	Identifier of the restriction area
linearElementSections	infra.LinearElementSection	1..*	Specific linear element sections
specificRestrictions	SpecificRestriction	1..*	List of restrictions

[SPT3TMS-11791]

Example: [SPT3TMS-12134]

Class "RestrictionArea"

{"structs": [
{
"name": "RestrictionArea",
"info": "Defines a topological area, in which the restrictions must be applied.",
"attrs": [
{"intId": 1, "name": "linearElementSections", "composition": "infra.LinearElementSection", "multiplicity": "1..*", "info": "composes of linear element sections"},
{"intId": 2, "name": "specificRestrictions", "composition": "SpecificRestriction", "multiplicity": "1..*", "info": "Defines the list of specific restrictions"},
{"intId": 3, "name": "id", "dataType": "string", "key": "global", "info": "Id of Restriction Area, used for referencing"}
]
}
]
} [SPT2TS-125318]

10.3.4 Class: SpecificRestriction

Description:

This class describes the characteristics of a specific restriction [SPT3TMS-12082]

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
appliedToTrains	TrainGroupSpec	0..1	if not provided the restriction is valid for all trains	-
restrictedAspects	RestrictedAspects	1..*	Restricted aspects	-

[SPT3TMS-12084 **]**

Example: [SPT3TMS-12131 **]**

Class SpecificRestriction

{"structs": [
{
"name": "SpecificRestriction",
"info": "Defines single aspect of the infrastructure restriction.",
"attrs": [
{"intId": 1, "name": "appliedToTrains", "composition": "TrainGroupSpec", "multiplicity": "0..1", "info": "Defines the applicable trains that are affected by the specific restriction"},
{"intId": 2, "name": "restrictedAspects", "composition": "RestrictedAspect", "multiplicity": "1..*", "info": "Defines the restricted aspects associated to a specific restriction"}
]
}]
} **[**SPT2TS-125315 **]**

10.3.5 Class: RestrictedAspects

Description:

This class describes the characteristics of a Restricted aspect. **[**SPT3TMS-12075 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
speedRestriction	SpeedRestriction	1	Speed restriction with add-on for ATO
allowedDrivingModes	OnboardOperatingMode	0..*	Train driving modes
nonStandstill	boolean	1	if true, trains specified inTrainGroupSpec cannot stop in the restriction area
trackClosure	boolean	1	If true, the trains defined in trainGroupSpec shall not enter restriction area
operationalRadioHole	boolean	1	If true radio communication is restricted within restriction area.
maxAdhesion	AdhesionCategory	1	Maximum Adhesion
atoInhibition	boolean	1	Train section is not continuously monitored, or stimulate driver's attention, or approaching overcrowded station.
maxCurrent	uint32 (A)	1	Electric current limitation in Amps, s. item 0.33 in SS126.
soundHorn	boolean	1	If true, approaching trains must sound horn
nonSwitchability	boolean	1	if true, Switchable Trackside asset cannot be operated
safetyCriticalCommandOnly	boolean	1	If true, only safety critical commandsare allowed (which are anyway set by Signal operator in PES)
manualOperationOnly	boolean	1	Only manual operation is allowed. This is a prescription for PES.
operationally/NonStandstill	boolean	1	Slightly different from the previous "nonStandstill" attribute. If true, the trains defined in trainGroupSpec cannot stop in the restriction area for operational reasons.

[SPT3TMS-12077 **]**

NOTE FOR MAINTENENCE: propose to add the following to the list of restricted aspects:

1. area without ETCS equipment,
2. area with no electric traction.

Example: [SPT3TMS-12126 **]**

Class RestrictedAspect

{"structs": [
{
"name": "RestrictedAspect",
"info": "Defines single aspects of the infrastructure restriction",
"union": true,
"attrs": [
{"intId": 1, "name": "speedRestriction", "composition": "SpeedRestriction", "info": "Defines the allowed speed"},
{"intId": 2, "name": "allowedDrivingModes", "enumType": "OnboardOperatingMode", "multiplicity": "*", "info": "Defines the allowed driving modes"},
{"intId": 3, "name": "nonStandstill", "dataType": "boolean", "info": "Defines the non stopping area. If true, the trains defined in trainGroupSpec shall not stop within restriction area."},
{"intId": 4, "name": "trackClosure", "dataType": "boolean", "info": "Defines track closure information. If true, the trains defined in trainGroupSpec shall not enter restriction area"},
{"intId": 5, "name": "operationalRadioHole", "dataType": "boolean", "info": "Defines the presence of operational radio holes. if true radio communication is restricted within restriction area"},
{"intId": 6, "name": "maxAdhesion", "enumType": "AdhesionCategory", "info": "Defines the maximum adhesion"},
{"intId": 7, "name": "atoInhibition", "dataType": "boolean", "info": "Defined the ato inhibition zones. Exmaples: train section is not contantly monitored, or stimulate driver's attention, or approaching overcrouded station"},
{"intId": 8, "name": "maxCurrent", "dataType": "uint32", "unit": "A", "info": "Defines max current value. For ATO, SS126. Ignored by TrafficCS"},
{"intId": 9, "name": "soundHorn", "dataType": "boolean", "info": "Defines that approaching trains needs to sound horn"},
{"intId": 10, "name": "nonSwitchability", "dataType": "boolean", "info": "Defines that Switchable Trackside Asset cannot be switched. To be clarified."},
{"intId": 11, "name": "safetyCriticalCommandOnly", "dataType": "boolean", "info": "Defines that only safety critical commands are allowed"},
{"intId": 12, "name": "manualOperationOnly", "dataType": "boolean", "info": "Defines that only manual operation by operator or maintainer is allowed (no automatic operation by PES)"},
{"intId": 13, "name": "operationallyNonStandstill", "dataType": "boolean", "info": "Defines the operationally non stopping area. If true, the trains defined in trainGroupSpec should not stop within restriction area due to operational reasons."}
]
}]
} **[**SPT2TS-125320 **]**

10.3.6 Class: SpeedRestriction

Description:

This class describes the characteristics of a speed restriction with an attribute which can be useful for ATO based features. **[**SPT3TMS-16524 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
maxSpeed (kmh)	uint32 in km/h	1	defines maximum allowed speed
restrictionEndsAtTrainFront	boolean	1	Normally set to "false". When "true", in case of disturbed level crossings the train can accelerate as soon as the train-front passed the level crossing.

[SPT3TMS-16526 **]**

10.3.7 Class: TrainGroupSpec

Description:

This class describes the train characteristics known to interlocking for safety relevant restrictions. **[**SPT3TMS-12066 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
axleLoadCategories	infra.LoadCapabilityLineCategories	0..*	Axle load categories
cantDeficiencyCategories	infra.CantDeficiencyCategory	0..*	Trains with cant deficiency lower than defined here shall not enter restriction area. Use CD_UNDEFINED if not active.

[SPT3TMS-12068 **]**

Example: [SPT3TMS-12127 **]**

Class "TrainGroupSpec"

{"structs": [
{
"name": "TrainGroupSpec",
"info": "Defines the train aspects, which define if the train belongs to the restricted group.",
"attrs": [
{"intId": 1, "name": "axleLoadCategories", "enumType": "infra.LoadCapabilityLineCategories", "multiplicity": "*", "info": "Trains with the listed Axle-Load-Category belong to the TrainGroup."},
{"intId": 2, "name": "cantDeficiencyCategories", "enumType": "infra.CantDeficiencies", "multiplicity": "*", "info": "Trains with the listed Cant Deficiency Category belong to the TrainGroup."}
]
}]
} **[**SPT2TS-125319 **]**

10.4 Operational Plan Warning Measure

10.4.1 Class: OperationalPlanWarningMeasure

Description:

This class defines the implementation of one or more Operational Plan Warning Measures. It relates to one or more Warning Areas and it contains references to the Warning Measure Events of these Warning Areas. **[**SPT3TMS-6642 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
id	string	1	Universally unique identifier used for identification of the object across systems.	Unique identifier for the communication across systems.
issuedAt	timestamp	1	Time issued at	-
configurationDataVersionRef	string	1	version of infrastructure data, which the Operational Plan refers to	-
warningMeasureEvents	Warning Measure Event	1..*	Sequence of Warning Measure Events to be implemented	Characteristics of Warning Measure Events are required for implementation and triggering of requests via SCI-CMD

[SPT3TMS-11793 **]**

Example: [SPT3TMS-12136 **]**

Class "Warning Measure"

{
"structs": [
{
"name": "OperationalPlanWarningMeasure",
"info": "Defines a temporary warning area, required for e.g. a possession.",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Defines the Identity of the object; used for referencing"},
{"intId": 2, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 3, "name": "configurationDataVersionRef", "dataType": "string", "info": "Defines the configuration data version for which the operational plan is valid"},
{"intId": 4, "name": "warningMeasureEvents", "composition": "WarningMeasureEvent", "multiplicity": "1..*", "info": "Defines the list of warning measures events"}
]
}]
}

[SPT2TS-124453 **]**

10.4.2 Class: WarningMeasureEvent

Description:

This class describes the characteristics of a Warning Measure Event. **[**SPT3TMS-14072 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
id	string	1	Universally unique identifier used for identification of the object across systems.	Unique identifier for the communication across systems.
creation Time	timestamp	1	Timestamp at which the Warning Area shall be created.	For reporting of updates via TMS <> CCS
removal Time	timestamp	1	Timestamp at which the Warning Area is planned to be removed.	For reporting of updates via TMS <> CCS
warning Area	Warning Area	1	Warning Areas to be implemented	Characteristics of Warning Area is required for implementation of requests via SCI-CMD
startsAfterEvents	OperationalEvent Ref	0..*	List of dependencies between this Operational Event and any other Operational Event of the same or another Operational Plan.	For implementation and triggering of requests via SCI-CMD.

[SPT3TMS-14074 **]**

Example: [SPT3TMS-14075 **]**

{"structs": [
{
"name": "WarningMeasureEvent",
"info": "Defines planned activation/removal of warning area",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Defines the Identity of the object; used for referencing"},
{"intId": 2, "name": "creationTime", "dataType": "timestamp", "info": "Defines the time when the warningArea shall be created=implemented=activated in interlocking, in UTC"},
{"intId": 3, "name": "removalTime", "dataType": "timestamp", "info": "Defines the removal time in UTC"},
{"intId": 4, "name": "warningArea", "composition": "WarningArea"},
{"intId": 5, "name": "startsAfterEvents", "composition": "OperationalEventRef", "multiplicity": "0..*", "info": "Defines the set of operational events, which must be finished before the current event starts."}
]

}]
} [SPT2TS-125324]

10.4.3 Class: WarningArea

Description:

This class describes the characteristics of a Warning Area. **[**SPT3TMS-6641 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
trackArea	infra.NetAreaReference	1..*	Track edge sections	-
warningDevices	WarningDevice	1..*	List of planned warning devices assigned to the Warning Area that shall be “READY_TO_WARN_IN_OWA” for an Operational Movement to enter the Warning Area. This attribute is subject to change, depending on the final decision whether Warning Devices shall be integrated in CCS.	For implementation of requests via SCI-CMD

[SPT3TMS-11794 **]**

Example: [SPT3TMS-12132 **]**

Class WarningArea

{"structs": [
{
"name": "WarningArea",
"info": "Defines area in which Warning measures shall be implemented.",
"attrs": [
{"intId": 1, "name": "trackArea", "composition": "infra.NetAreaReference", "info": "protected area"},
{"intId": 2, "name": "warningDevices", "composition": "WarningDevice", "multiplicity": "1..*", "info": "Defines the list of warning devices"}
]
}]
} **[**SPT2TS-125322 **]**

10.4.4 Class: WarningDevice

Description:

This class describes the characteristics of a Warning Device. **[**SPT3TMS-12085 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:	Reason:
id	string	1	Universally unique identifier used for identification of the object across systems.	Unique identifier for the communication across systems.
deviceType	Device Type	1	Defines the different warning devices	-

[SPT3TMS-12087 **]**

Example: [SPT3TMS-12130 **]**

Class WarningDevice

{"structs": [
{
"name": "WarningDevice",
"info": "Defines functional warning device which must be activated to implement warning measure",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Defines the Identity of the object; used for referencing"},
{"intId": 2, "name": "deviceType", "enumType": "DeviceType", "info": "Defines a functional warning device type used for securing the warning area"}
]
}]
} **[**SPT2TS-125323 **]**

10.4.4.1 Enum: DeviceType

Description:

This enum describes the possible device types. **[**SPT3TMS-12088 **]**

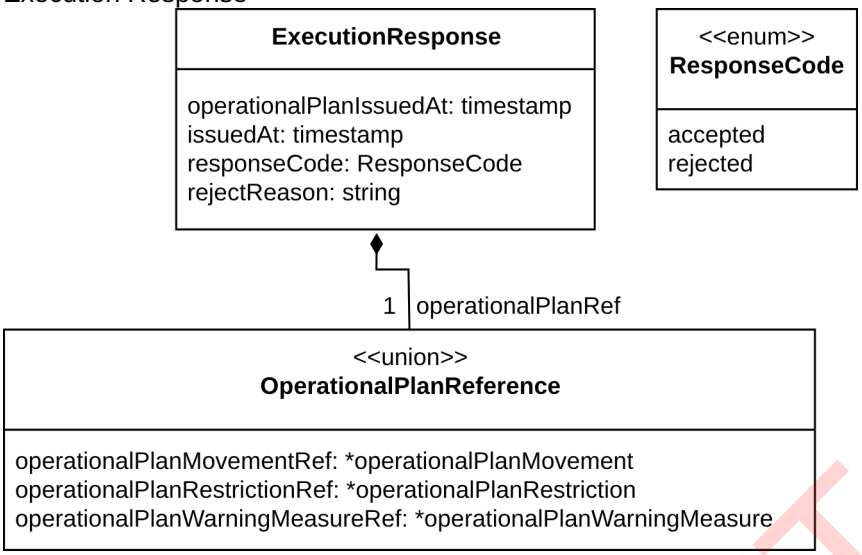
Attributes:

Name:	Description:
acousticalIndicator	Acoustical indicator
opticalIndicator	Optical indicator
hapticIndicator	Haptic indicator

[SPT3TMS-12091 **]**

10.5 Operational Plan Execution Response

Execution Response



[SPT2TS-125684]

Figure 22 : Class diagram for Operational Plan Operational Response

10.5.1 Class: ExecutionResponse

Description:

This class describes the Execution Response. [SPT3TMS-14077]

Attributes:

Name:	Type:	Multiplicity:	Description:
operationalPlanRef	OperationalPlanReference	1	Plan Identifier
operationalPlanIssuedAt	timestamp	1	Plan issued at time.
issuedAt	timestamp	1	Response time.
responseCode	ResponseCode	1	Code indicating the acceptance or the reason in case of a rejection.
rejectReason	string	0..1	Description of rejection

[SPT3TMS-14080]

Example: [SPT3TMS-14079]

Class ExecutionResponse
{

"structs": [
{
"name": "ExecutionResponse",
"info": "Defines the result of evaluation of an operational plan from the TrafficCS (accepts/rejects)",
"attrs": [
{"intId": 1, "name": "operationalPlanRef", "composition": "OperationalPlanReference", "info": "Refers to an operational plan"},
{"intId": 2, "name": "operationalPlanIssuedAt", "dataType": "timestamp", "info": "refers to the issuedAt-value of the operational plan request, in UTC"},
{"intId": 3, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 4, "name": "responseCode", "enumType": "ResponseCode", "info": "Defines the respose code associated with the event execution request"},
{"intId": 5, "name": "rejectReason", "dataType": "string", "multiplicity": "0..1", "info": "Defines the reject reason for an event execution request"}
]
}]
} **[**SPT2TS-125683 **]**

10.5.2 Class: OperationalPlanReference

Description:

This class defines the implementation of an Operational Plan Identifier. **[**SPT3TMS-12381 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
operationalPlanMovementRef	OperationalPlanMovement	0..1	Message for movements
operationalPlanRestrictionRef	OperationalPlanRestriction	0..1	Message for restriction
operationalPlanWarningMeasureRef	OperationalPlanWarningMeasure	0..1	Message for warningMeasure

[SPT3TMS-12383 **]**

Example: [SPT3TMS-12411 **]**

Class PlanIdentifier

{
"structs": [
{
"name": "OperationalPlanReference",
"info": "Defines a reference to one of possible OperationalPlans (movement, restriction, warning)",
"union": true,
"attrs": [
{"intId": 1, "name": "operationalPlanMovementRef", "reference": "OperationalPlanMovement", "info": "Refers the operational plan movement"},
{"intId": 2, "name": "operationalPlanRestrictionRef", "reference": "OperationalPlanRestriction", "info": "Refers the operational plan restriction"},
{"intId": 3, "name": "operationalPlanWarningMeasureRef", "reference": "OperationalPlanWarningMeasure", "info": "Refers the operational plan warning measure"}
]
}
],
"enums": [
{
"name": "ResponseCode",
"enumLiterals": [
{"intId": 0, "name": "accepted"},
{"intId": 1, "name": "rejected"}
]
}
]
}

[SPT2TS-125685 **]**

10.5.3 Enum: ResponseCode

Description:

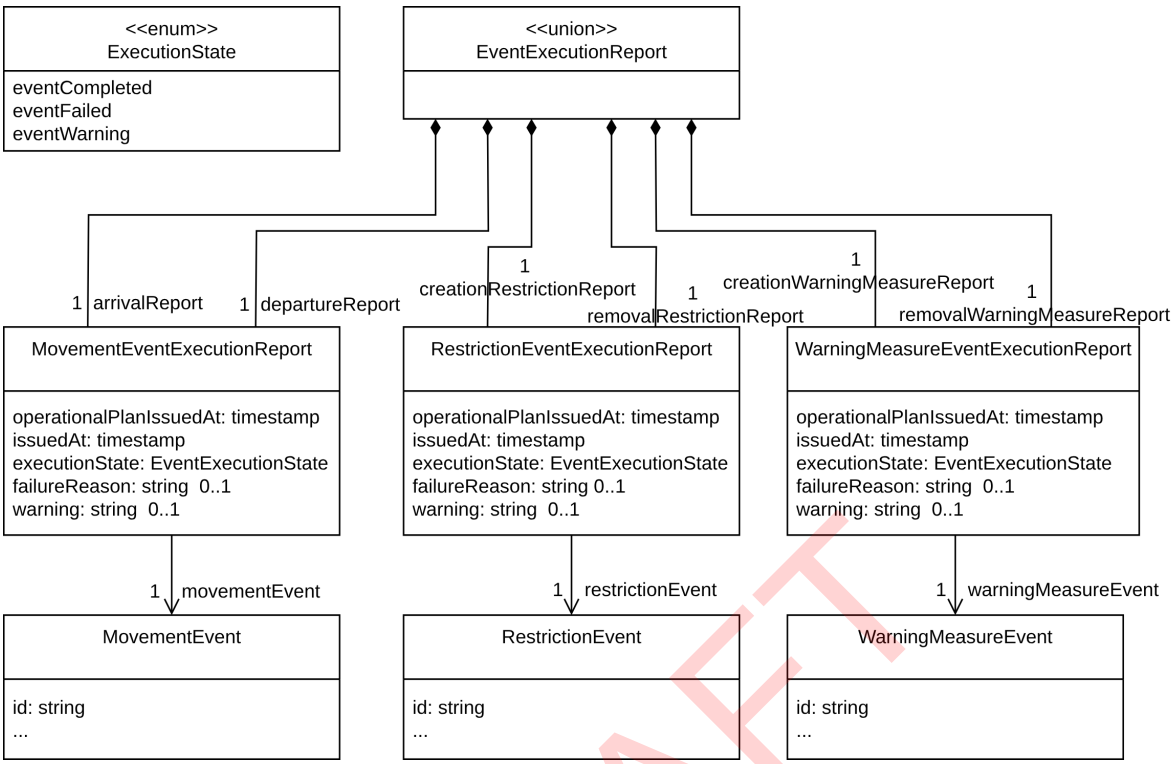
This enum describes possible response codes. **[**SPT3TMS-12384 **]**

Attributes:

Name:	Description:
accepted	Execution Request of Operational Plan was accepted
rejected	Execution Request of Operational Plan was rejected The Operational Plan Feasibility Check and its details will be described here in a later version of this document: Details of the possible causes to be defined but a preliminary list is defined here meanwhile: 1) Forced reduction of track speed 2) Unforeseen extension of stop time at station due to a sudden technical fault 3) ...

[SPT3TMS-12380 **]**

10.6 Operational Plan Execution Report



[SPT2TS-127059 **]**

Figure 23 : Class diagram for Operational Plan Execution Report

10.6.1 Class: EventExecutionReport

Description:

The class describes the Execution State of a referenced Operational Event from Plan Execution System or ATO Trackside perspective. **[**SPT3TMS-14081 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
arrivalReport	MovementEventExecutionReport	0..1	For passing only the departure report shall be sent
departureReport	MovementEventExecutionReport	1	Departure Report
creationRestrictionReport	RestrictionEventExecutionReport	1	Creation Restriction Report.
removalRestrictionReport	RestrictionEventExecutionReport	1	Removal Restriction Report.
		1	Creation Warning Measure Report.

Name:	Type:	Multiplicity:	Description:
creationWarningMeasureReport	WarningMeasureEventExecutionReport		
removalWarningMeasureReport	WarningMeasureEventExecutionReport	1	Removal Warnining Measure Report.

[SPT3TMS-14083 **]**

Example: [SPT3TMS-14084 **]**

{
"structs": [
{
"name": "EventExecutionReport",
"info": "Defines states of start/end of an event (success/failure, timestamp etc.)",
"union": true,
"attrs": [
{"intId": 1, "name": "arrivalReport", "composition": "MovementEventExecutionReport", "info": "Defines the arrival reports. For passing only the departure reports are to be sent"},
{"intId": 2, "name": "departureReport", "composition": "MovementEventExecutionReport", "info": "Defines the result of departure attempt"},
{"intId": 3, "name": "creationRestrictionReport", "composition": "RestrictionEventExecutionReport", "info": "Defines the result of creation of a restriction"},
{"intId": 4, "name": "removalRestrictionReport", "composition": "RestrictionEventExecutionReport", "info": "Defines the result of removal of a restriction"},
{"intId": 5, "name": "creationWarningMeasureReport", "composition": "WarningMeasureEventExecutionReport", "info": "Defines the result of creation of a warning measure"},
{"intId": 6, "name": "removalWarningMeasureReport", "composition": "WarningMeasureEventExecutionReport", "info": "Defines the result of removal of warning measure"}
]
},
{
"name": "MovementEventExecutionReport",
"info": "Defines state of execution of a warning measure event (completed, failed, warning).",
"attrs": [
{"intId": 1, "name": "movementEvent", "reference": "MovementEvent", "info": "refers to a movement event. In json is a string with planId/eventId"},
{"intId": 2, "name": "operationalPlanIssuedAt", "dataType": "timestamp", "info": "in UTC"},
{"intId": 3, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 4, "name": "executionState", "enumType": "EventExecutionState", "info": "Defines the type of execution result of the event"},
{"intId": 5, "name": "failureReason", "dataType": "string", "multiplicity": "0..1", "info": "Defines an informal error message in English"},
{"intId": 6, "name": "warning", "dataType": "string", "multiplicity": "0..1", "info": "Defines an informal warning message in English"}
]
},
{
"name": "RestrictionEventExecutionReport",
"info": "Defines how the restriction event was executed (success, failure ...)",
"attrs": [
{"intId": 1, "name": "restrictionEvent", "reference": "RestrictionEvent", "info": "refers to a restriction event including operational plan restriction"},
{"intId": 2, "name": "operationalPlanIssuedAt", "dataType": "timestamp", "info": "in UTC"},

{"intId": 3, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 4, "name": "executionState", "enumType": "EventExecutionState", "info": "Defines the type of execution result of the event"},
{"intId": 5, "name": "failureReason", "dataType": "string", "multiplicity": "0..1", "info": "Defines an informal error message in English"},
{"intId": 6, "name": "warning", "dataType": "string", "multiplicity": "0..1", "info": "Defines an informal warning message in English"}
]
},
{
"name": "WarningMeasureEventExecutionReport",
"attrs": [
{"intId": 1, "name": "warningMeasureEvent", "reference": "WarningMeasureEvent", "info": "in json is a string with planId/eventId"},
{"intId": 2, "name": "operationalPlanIssuedAt", "dataType": "timestamp", "info": "in UTC"},
{"intId": 3, "name": "issuedAt", "dataType": "timestamp", "info": "iDefines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 4, "name": "executionState", "enumType": "EventExecutionState", "info": "Defines the type of execution result of the event"},
{"intId": 5, "name": "failureReason", "dataType": "string", "multiplicity": "0..1", "info": "Defines an informal error message in English"},
{"intId": 6, "name": "warning", "dataType": "string", "multiplicity": "0..1", "info": "Defines an informal warning message in English"}
]
}
],
"enums": [
{
"name": "EventExecutionState",
"enumLiterals": [
{"intId": 0, "name": "eventCompleted"},
{"intId": 1, "name": "eventFailed"},
{"intId": 2, "name": "eventWarning"}
]
}]
} **[**SPT2TS-125664 **]**

10.6.2 Class: MovementEventExecutionReport

Description:
The class describes the Execution State of a referenced Operational Event from PES's perspective. **[**SPT3TMS-14086 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
movementEvent	MovementEvent	1	Movement Event
operationalPlanIssuedAt	timestamp	1	Plan Issued Time
issuedAt	timestamp	1	Report Time
executionState	EventExecutionState	1	State of Execution
failureReason	string	0..1	Informal error message in English
warning	string	0..1	Informal warning message in English

[SPT3TMS-14088 **]**

10.6.3 Class: RestrictionEventExecutionReport

Description:
The class describes the Execution State of a referenced Operational Event from PES's perspective. **[**SPT3TMS-14091 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
restrictionEvent	RestrictionEvent	1	Restriction Event
operationalPlanIssuedAt	timestamp	1	Plan Issued Time
issuedAt	timestamp	1	Report Time
executionState	EventExecutionState	1	State of Execution
failureReason	string	0..1	Informal error message in English
warning	string	0..1	Informal warning message in English

[SPT3TMS-14098 **]**

10.6.4 Class: WarningMeasureEventExecutionReport

Description:

The class describes the Execution State of a referenced Operational Event from PES's perspective. **[**SPT3TMS-14090 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
warningMeasureEvent	WarningMeasureEvent	1	Warning Measure Event
operationalPlanIssuedAt	timestamp	1	Plan Issued Time
reportTime	timestamp	1	Report Time
executionState	EventExecutionState	1	State of Execution
failureReason	string	0..1	Informal error message in English
warning	string	0..1	Informal warning message in English

[SPT3TMS-14099 **]**

10.6.4.1 Enum: EventExecutionState

Description:

This enum describes possible execution states. **[**SPT3TMS-14095 **]**

Attributes:

Name:	Description:
eventCompleted	The Operational Event has been successfully completed.
eventFailed	The event execution failed.
eventWarning	The event execution issued a warning

[SPT3TMS-14097 **]**

10.7 Train Unit Report PE

10.7.1 Class: TrainUnitReportPE

Description:

This class describes the ETCS specific information of an identified Physical Train Unit provided by Plan Execution System. **[**SPT3TMS-12367 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
issuedAt	timestamp	1	Time issued at
operationalPlanMovement	OperationalPlanMovement	0..1	Operational Plan Movement
cmdTrainObject	TrainObject	1	Train Object

[SPT3TMS-12366 **]**

Example: [SPT3TMS-13982 **]**

{
"structs": [
{
"name": "TrainUnitReportPE",
"info": "Defines train report (position, capabilities, permission) coming from PlanExecution system.",
"attrs": [
{"intId": 1, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 2, "name": "operationalPlanMovement", "reference": "OperationalPlanMovement", "multiplicity": "0..1", "info": "Refers to the operational plan movement. If the new train was created in TrafficCS it does not have an operationalPlan (Movement) yet"},
{"intId": 3, "name": "cmdTrainObject", "composition": "TrainObject", "info": "Defines the train object information"}
]
}]} **[**SPT2TS-125679 **]**

10.7.2 Class: TrainObject

Description:

This class describes the train object **[**SPT3TMS-14199 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	Identifies uniquely the Train Object
obuIdentifier	string	1	OBU Identifier; it represents the NID-ENGINE
movementPermission	MovementPermission	0..1	Movement permission
safeTrainLength	boolean	1	If true, indicates that the value which reports the train length is trustable and valid
communicationActive	boolean	1	If true, the communication is active?
trainRunningNumber	string	1	Train running number; it represents the NID-OPERATIONAL
onboardOperatingMode	OnboardOperatingMode	1	Onboard operating Mode
safeTrainExtent	infra.NetLinearReference	1	Reports the safe train extent as given by ETPS
maRequestedByTrain	MaRequestedByTrain	1	Ma-Requested by Train
speed	uint32 (kmh)	1	Current speed in km/h
validatedTrainData	ValidatdeddTrainData	1	Provides a set of static and quasi-static data which characterize the train and change very rarely
runningDirection	RunningDirection	1	Indicates the running direction of the train
axleLoadCategory	infra..LoadCapabilityLineCategories	1	Axle load category
trainLength	uint32	1	Train length in meter
vMaxTrain	uint32	1	vMax Train in km/h
trainLocation	infra.NetLinearReference	1	Contains the position of the entire train (i.e., from rear to front) including the confidence intervals
cantDeficiencyCategory	infra.CantDeficiencies	1	Cant deficiency category

[SPT3TMS-14198 **]**

Class TrainObject
{"structs": [
{
"name": "TrainObject",
"info": "Defines a train report generated by TrafficCS including its position, length, speed etc.",
"attrs": [
{"intId": 1, "name": "obuIdentifier", "dataType": "string", "info": "Defines the onboard-unit identifier, see ss026, 7.5.1.88 NID_ENGINE"},
{"intId": 3, "name": "movementPermission", "composition": "MovementPermission", "multiplicity": "0..1", "info": "Movement Permission associated to Train Object"},
{"intId": 5, "name": "safeTrainLength", "dataType": "boolean", "info": "Defines if the trainLength-value is safe, see ss026, 7.5.1.112 Q_LENGTH"},
{"intId": 6, "name": "communicationActive", "dataType": "boolean", "info": "Defines the communication state for the train object, see ss026, 7.5.1.127 Q_RBC"},
{"intId": 7, "name": "trainRunningNumber", "dataType": "string", "info": "see PhysicalTrainUnitIdentifier.trainRunningNumber; see ss026, 7.5.1.92 NID_OPERATIONAL"},
{"intId": 9, "name": "onboardOperatingMode", "enumType": "OnboardOperatingMode", "info": "Defines the on-board operating mode of the train, see ss026, 7.5.1.72 M_MODE"},
{"intId": 10, "name": "safeTrainExtent", "composition": "infra.NetLinearReference", "info": "train extent starting with the train head (opposite to the train-movement-direction); defines the safe train extent as provided by ETPS"},
{"intId": 11, "name": "id", "dataType": "string", "key": "global", "info": "Id of Train Object generated by ETPS; used for referencing"},
{"intId": 12, "name": "maRequestedByTrain", "enumType": "MaRequestedByTrain", "info": "see ss026, 7.5.1.118.3 Q_MARQSTREASON"},
{"intId": 13, "name": "speed", "dataType": "uint32", "unit": "km/h", "info": "Defines current speed of the train, see ss026, 7.5.1.172 V_TRAIN"},
{"intId": 14, "name": "validatedTrainData", "composition": "ValidatedTrainData", "info": "see ss026, 7.4.3.5 Packet Number 11: Validated train data"},
{"intId": 15, "name": "runningDirection", "enumType": "RunningDirection", "info": "see ss026, 7.5.1.105 Q_DIRTRAIN"}
]
}]} **[**SPT2TS-131447 **]**

10.7.3 Class: ValidatedTrainData

Description:

This class qualifies the train with a set of attributes which are specific for that train and are necessary to apply TMS algorithms. The don't need to be transmitted every time, as they don't change at every train movement. **[**SPT3TMS-16955 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
cantDeficiencyCategory	infra.CantDeficiencies	1	Reports the category of the cant deficiency. See SS126, NC_CDTRAIN
	uint32	1	Indicates other train category

Name:	Type:	Multiplicity:	Description:
otherTrainCategory			
trainLength	uint32 (m)	1	Train length in meters
vMaxTrain	uint32 (kmh)	1	Maximum speed of the train in km/h
loadingGauge	infra.LoadingGaugeType	1	See SS126, M_LOADINGGAUGE
axleLoadCategory	infra.LoadCapabilityLineCategories	1	Axle load category
airtight	AirtightType	1	See SS126, M_AIRTIGHT
axlenumber	uint32	1	Number of axles of the train unit related to the Onboard equipment
voltage	VoltageSystem	0..*	Indicates the voltage system
nidNtc	uint32	1..*	See SS126, NID-NTC

[SPT3TMS-16956 **]**

Maintenance activity: add here the link to OPP .json description of this class

10.7.4 Enum: VoltageSystem

Description:

This enumerated lists the possible voltage systems **[**SPT3TMS-16958 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
voltage	infra.EnergySupplySystems	1	See SS126, M_VOLTAGE
countryTractionId	uint32	1	See SS126, NID_CTRACTION

[SPT3TMS-16959 **]**

Maintenance activity: add here the link to OPP .json description of this class

10.7.5 Class: MovementPermission

Description:

This class describes the characteristics of the movement permission, in terms of speed profile and path. **[**SPT3TMS-13985 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	id of the movement permission
riskBuffer	infra.NetLinearReference	1	overlap after MP
riskPaths	infra.NetLinearReference	*	Risk paths; defines flank protection related to neighbouring paths
mpExtent	infra.NetLinearReference	1	Extent of the MP
mpOnboardOperatingModes	MpOnboardOperatingMode	0..*	Reports the mode profile applied
mpSpeeds	MpSpeed	1..*	Speed profiles
excludedSTAForFlankProtection	SwitchableTracksideAsset	*	List of Switchable Trackside Assets excluded from flank protection
currentEoA	infra.NetLinearReference	1..*	Current EoA
pendingCooperativeShorteningRequests	CooperativeShorteningRequest	0..*	Pending requests for cooperative shortening of Movement Authority

[SPT3TMS-13987]

Maintenance activity: add here the link to OPP .json description of this class

10.7.6 Class: MPSpeed

Description:

This class extends the characteristics of a maximum speed, associating it to the applicable path.
[SPT3TMS-16965]

Attributes:

Name:	Type:	Multiplicity:	Description:
vMax	int	1	Maximum applicable speed
linkedPath	infra.NetLinearReference	1	Path which the speed applies to

[SPT3TMS-16964]

10.7.7 Class: MPOnboardOperatingMode

Description:

This class descrfibes the driving mode along a defined path. **[**SPT3TMS-13994 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
onboardOperatingMode	OnboardOperatingMode	1	Driving mode to be applied along to a specific path
path	infra.NetLinearReference	1	Path which the driving mode applies to

[SPT3TMS-13997 **]**

10.7.8 Class: MpSpeed

Description:

This class describes the Mp Speed. **[**SPT3TMS-13993 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
vMax	uint32	1	
path	infra.NetLinearReference	1	

[SPT3TMS-13992 **]**

10.7.9 Enum: OnboardOperatingMode

Description:

This enum describes the onboard operating mode. **[**SPT3TMS-13988 **]**

Attributes:

Value:	Description:
fullSupervision	Full supervision mode
onSight	On sight mode
staffResponsible	Staff Responsible mode
shunting	Shunting mode mode
infitted	Infitted mode mode
sleeping	Sleeping mode mode
standBy	StandBy mode
trip	Trip mode
postTrip	Post-Trip mode
systemFailure	System Failure
isolation	Isolation
nonLeading	Non-Leading mode
limitedSupervision	Limited Supervision mode
nationalSystem	National System
reversing	Reversing mode
passiveShunting	Passive Shunting mode

Value:	Description:
automaticDriving	Automatic Driving mode
superviseManouvre	Supervise Manouvre mode

[SPT3TMS-13991]

10.7.10 Enum: RunningDirection

Description:

This enum describes the possible running directions for a train.

Attributes:

Value:	Description:
reverse	The train runs along the reverse direction
nominal	The train runs along the nominal direction
unknown	direction unknown

10.7.11 Enum: MaRequestedByTrain

Description:

This enum describes the possible ways by which a train requests a Movement Authority.
[SPT3TMS-16970]

Attributes:

Value:	Description:
startSelectedByDriver	Request forced by the driver
timeBeforeReachingThePerturbationLocationReached	Request originatd by the system a configured time before the estimated arrival to a perturbed location
timeBeforeASectionTimer_LoaSpeedTimerExpiresReached	**Maintenance activity:** to be clarified with TCSteam (MK)
trackDescriptionDeleted	**Maintenance activity:** to be clarified with TCSteam (MK)
tafUpToLevel2TrainsitionLocation	Request originatd by a TAF to Level 2 transition

[SPT3TMS-16972]

Maintenance activity: add here the link to OPP .json description of this class

10.7.12 Enum: Airtight

Description:

This enum describes whether a train is equipped with airtight or not. **[**SPT3TMS-16973 **]**

Attributes:

Name:	Description:
NOT_FITTED	The train is not fitted with airtight
FITTED	he train is fitted with airtight

[SPT3TMS-16975 **]**

10.7.13 Class: CooperativeShorteningRequest

Description:

This class has yet to be defined. **[**SPT3TMS-16966 **]**

Attributes:

Name:	Description:
undefined	Undefined

[SPT3TMS-16968 **]**

10.8 Train Unit Report Ato Status

Description:
This class describes an identified Physical Train Unit provided by ATO Trackside based on the ATO Status Report (AoE SS-126). **[**SPT3TMS-14207 **]**

10.8.1 Class: TrainUnitReportAtoStatus

Description:
This class describes the data of an identified Physical Train Unit provided by ATO Trackside based on the ATO Status Report (ATO over ETCS (AoE) SS-126). The trainLength will be compared with the scheduled train length (input from TSI), provided by TMS. **[**SPT3TMS-12369 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
issuedAt	timestamp	1	Time issued at
operationalPlanMovement	OperationalPlanMovement	0..1	If the new train was created in TrafficCS it does not have an operationalPlan (Movement) yet
trainUnitIdentifier	PhysicalTrainUnitIdentifier	1	Train Unit identifier
atoState	AtoState	1	ATO state
routingError	boolean	1	inconsistent path
nextStoppingPointSkip	boolean	1	Skip-request by the train driver.
lowAdhesionReportedByDriver	boolean	1	Low adhesion reported by driver
operationalConditionsFulfillment	boolean	1	OperationalConditionsFulfillment
isMoving	boolean	1	Is moving?
unableToStopAtNextStoppingPoint	boolean	1	The train is too close and/or too fast for stopping
isSlipping	boolean	1	Is slipping?
speed	uint32	1	Speed in km/h
trainLength	uint32	1	Length in meter
driverId	string	0..1	Driver ID
frontPosition	infra.NetPointReference	1	The direction can be deduced from JP and STR in SS 126
eventHandling	EventHandling	0..1	Event handling
eventEstimations		*	Event estimation

Name:	Type:	Multiplicity:	Description:
	EventEstimation		

[SPT3TMS-12368 **]**

Example: [SPT3TMS-12614 **]**

{
"structs": [
{
"name": "TrainUnitReportAtoStatus",
"info": "Defines train position report from ATO with additional info (driverId, speed, etc.). Source is SS126.STR.",
"attrs": [
{"intId": 1, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 2, "name": "operationalPlanMovement", "reference": "OperationalPlanMovement", "multiplicity": "0..1", "info": "Refers to the operational plan movement. If the new train was created in TrafficCS it does not have an operationalPlan (Movement) yet"},
{"intId": 3, "name": "trainUnitIdentifier", "composition": "PhysicalTrainUnitIdentifier"},
{"intId": 4, "name": "atoState", "enumType": "AtoState", "info": "Defines the current ATO On-Board state"},
{"intId": 5, "name": "routingError", "dataType": "boolean", "info": "Defines if a routing error has occured (inconsistent path)"},
{"intId": 6, "name": "nextStoppingPointSkip", "dataType": "boolean", "info": "Defines the wish of the train driver to skip the next stop. "},
{"intId": 7, "name": "lowAdhesionReportedByDriver", "dataType": "boolean", "info": "Defines any low adhesion is reported by the driver"},
{"intId": 8, "name": "operationalConditionsFulfillment", "dataType": "boolean", "info": "Defines if the operational conditions are fullfilled for ATO"},
{"intId": 9, "name": "isMoving", "dataType": "boolean", "info": "Defines if the train is moving or not (v != 0)"},
{"intId": 10, "name": "unableToStopAtNextStoppingPoint", "dataType": "boolean", "info": "Defines if the the train is too close and/or too fast for stopping"},
{"intId": 11, "name": "isSlipping", "dataType": "boolean", "info": "Defines if the wheel slip occurs. bit7 from SS126"},
{"intId": 12, "name": "speed", "dataType": "uint32", "unit": "km/h", "info": "current speed of the train"},
{"intId": 13, "name": "trainLength", "dataType": "uint32", "unit": "m", "info": "Defines the train length", "ontology": {"subPropertyOf": "http://data.europa.eu/949/length"}},
{"intId": 14, "name": "driverId", "dataType": "string", "multiplicity": "0..1", "info": "Defines the driver id for the train unit"},
{"intId": 15, "name": "frontPosition", "composition": "infra.NetPointReference", "multiplicity": "0..1", "info": "Defines the train front position with respect to the track edge direction. This direction can be deduced from JP and STR in SS 126. Optional as SS126 allows OBU to send 'undefined'"},
{"intId": 16, "name": "eventHandling", "composition": "EventHandling", "multiplicity": "0..1", "info": "defines how the previous event was handled"},
{"intId": 17, "name": "eventEstimations", "composition": "EventEstimation", "multiplicity": "*", "ordered": "byIndex", "info": "Refers to event estimations"}
]
},

{
"name": "EventHandling",
"info": "Defines, how the previous event was handled.",
"attrs": [
{"intId": 1, "name": "event", "reference": "MovementEvent", "info": "refers a movement event including the movement plan"},
{"intId": 2, "name": "passStopDepart", "enumType": "MovementEventAction", "info": "Defines the result of the event handling action"}
]
},
{
"name": "EventEstimation",
"info": "Defines forecast for events coming from ATO-Onboard",
"attrs": [
{"intId": 1, "name": "event", "reference": "MovementEvent", "info": "refers a movement event including the movement plan"},
{"intId": 2, "name": "arrival", "dataType": "timestamp", "info": "Defines the arrival time in UTC"}
]
}
],
"enums": [
{
"name": "MovementEventAction",
"enumLiterals": [
{"intId": 0, "name": "passedEvent"},
{"intId": 1, "name": "stoppedAccurateAtEvent"},
{"intId": 2, "name": "stoppedWithOvershootAtEvent"},
{"intId": 3, "name": "stoppedWithUndershootAtEvent"},
{"intId": 4, "name": "departedFromEvent"}
]
},
{
"name": "AtoState",
"enumLiterals": [
{"intId": 0, "name": "unknown"},
{"intId": 1, "name": "CO", "info": "State Configuration"},
{"intId": 2, "name": "NA", "info": "State Not Available"},
{"intId": 3, "name": "AV", "info": "State Available"},
{"intId": 4, "name": "RE", "info": "State Ready"},
{"intId": 5, "name": "EG", "info": "State Engaged"},
{"intId": 6, "name": "DE", "info": "State Disengaged"},
{"intId": 7, "name": "FA", "info": "State Failure"}
]
}
]
} **[**SPT2TS-125680 **]**

10.8.2 Class: DoorData

Description:

This class describes position and state of a single door of an identified Physical Train Unit provided by a Train Unit Report. **[**SPT3TMS-9759 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
doorState	DoorState	1	Left/right refers to the orientation of the Physical Train Unit. Source: TCR
lastPassengerExchangeTime	uint32 (s)	1	last passenger exchange time in secoinds. 65535 if unknown

[SPT3TMS-12372 **]**

Example: [SPT3TMS-12425 **]**

Class DoorData

{
"structs": [
{
"name": "DoorData",
"info": "Defines availability of a door and last open time.",
"attrs": [
{"intId": 1, "name": "doorState", "enumType": "DoorState", "info": "Defines the door state"},
{"intId": 2, "name": "lastPassengerExchangeTime", "dataType": "uint32", "unit": "s", "info": "Defines the last passenger exchange time. Use 65535 if unknown"}
]
}]
} **[**SPT2TS-125695 **]**

10.8.3 Class: AdditionalTrainData

Description:

This class describes the brakes of an identified Physical Train Unit provided by a Train Unit Report. **[**SPT3TMS-9760 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
maxAvailableTractiveEffort	uint32 (N)	1	Includes both multiple traction and reduced traction capabilities (isolated bogie etc.), in Newton Source: M_TCMS_Fmax (via SS-139 and TCR) · Range: 0 ... 3000 kN · Resolution 1 kN · Special values: o 3001 ... MAXNUM - 1: spare o MAXNUM: Unknown
maxAvailableTractivePower	uint32 (W)	1	Includes both multiple traction and reduced traction capabilities (isolated bogie etc.), in Watts Source: M_TCMS_Pmax (via SS-139 and TCR) · Range: 0 ... 32 000 kW · Resolution 1 kW · Special values: o 32001 ... MAXNUM - 1: spare o MAXNUM:Unknown
maxAvailableDynamicBrakeForce	uint32 (N)	1	Source: M_TCMS_FmaxDB (via SS-139 and TCR), in Newton · Range: 0 ... 3000 kN · Resolution 1 kN · Special values: o 3001 ... MAXNUM - 1: spare o MAXNUM:: Unknown
maxAvailableDynamicBrakePower	uint32 (W)	1	Source: M_TCMS_PmaxDB (via SS-139 and TCR), in Watts · Range: 0 ... 32 000 kW · Resolution 1 kW · Special values: o 32001 ... MAXNUM - 1: spare o MAXNUM:: Unknown

Name:	Type:	Multiplicity:	Description:
maxAvailableServiceBrakeForce	uint32 (N)	1	Source: M_TCMS_FmaxSB (via SS-139 and TCR), in Newton · Range: 0 ... 3000 kN · Resolution 1 kN · Special values: o 3001 ... MAXNUM - 1: spare o MAXNUM:: Unknown / not used
regenerativeBrakeState	ActiveState	1	Source: TCR
magneticShoeBrakeState	ActiveState	1	Source: TCR
eddyCurrentBrakeState	ActiveState	1	Source: TCR
electroPneumaticBrake State	ActiveState	1	Source: TCR
adhesionIndependentBrakeState	ActiveState	1	Source: via SS-034 and TCR
availableAdhesionCategory	AdhesionCategory	1	053 in Subset
availableAdhesionPhysicalValue	uint32	1	600 if unknown

[SPT3TMS-12374 **]**

Example: [SPT3TMS-12416 **]**

Class AdditionalTrainData

{
"structs": [
{
"name": "AdditionalTrainData",
"info": "Defines dynamic train characteristics, which are not primary for the timetable (therefore 'additional')",
"attrs": [
{"intId": 1, "name": "maxAvailableTractiveEffort", "dataType": "uint32", "unit": "N", "exp": 3, "range": "0..3000", "info": "Defines the maximum available traction effort. Use 65535 if unknown"},
{"intId": 2, "name": "maxAvailableTractivePower", "dataType": "uint32", "unit": "W", "exp": 3, "range": "0..32000", "info": "Defines the maximum available traction power. Use 65535 if unknown"},
{"intId": 3, "name": "maxAvailableDynamicBrakeForce", "dataType": "uint32", "unit": "N", "exp": 3, "range": "0..3000", "info": "Defines the maximum available dynamic brake force. Use 65535 if unknown"},
{"intId": 4, "name": "maxAvailableDynamicBrakePower", "dataType": "uint32", "unit": "W", "exp": 3, "range": "0..32000", "info": "Defines the maximum available dynamic brake power. Use 65535 if unknown"},
{"intId": 5, "name": "maxAvailableServiceBrakeForce", "dataType": "uint32", "unit": "N", "exp": 3, "range":

"0..3000", "info": "Defines the maximum available service brake force. Use 65535 if unknown"},
{"intId": 6, "name": "regenerativeBrakeState", "enumType": "ActiveState", "info": "Defines the state of the regenerative brake"},
{"intId": 7, "name": "magneticShoeBrakeState", "enumType": "ActiveState", "info": "Defines the state of magnetic shoe brake "},
{"intId": 8, "name": "eddyCurrentBrakeState", "enumType": "ActiveState", "info": "Defines the state of eddy current brake"},
{"intId": 9, "name": "electroPneumaticBrakeState", "enumType": "ActiveState", "info": "Defines the state of electro pneumatic brake"},
{"intId": 10, "name": "adhesionIndependentBrakeState", "enumType": "ActiveState", "info": "Defines the adhesion independent brake state"},
{"intId": 11, "name": "availableAdhesionCategory", "enumType": "AdhesionCategory", "info": "Defines the available adhesion categories; 053 in Subset"},
{"intId": 12, "name": "availableAdhesionPhysicalValue", "dataType": "uint32", "exp": -3, "info": "Defines the available adhesion value. 600 if unknown"}
]
}]
} **[**SPT2TS-125697 **]**

10.8.3.1 Enum: ActiveState

Description:

This enum describes possible active states. **[**SPT3TMS-12600 **]**

Attributes:

Name:	Description:
activeState	Active State
inactiveState	Inactive state
undefinedState	Undefined state

[SPT3TMS-12602 **]**

10.8.4 Class: PhysicalConsist

Description:

This class describes the Physical Consist of an identified Physical Train Unit provided by a Train Unit Report. **[**SPT3TMS-9761 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
evnIdentifier	uint32	1	European vehicle number (EVN) 12 digits, 0=unknown
passengerLoad	uint32 (%)	1	Passenger load in percent
mass	uint32 (kg)	1	Mass of Physical Consist (kg) Source: TCR • Range: 0 ... 150000kg Special values: • 15000.1... MAXNUM - 0.1: spare • MAXNUM: Unknown
maxCurrentConsuption	uint32 (A)	1	Max current consuption in Amps
physicalVehicle	PhysicalVehicle	0..255	List of Physical Vehicles of the Physical Consist Source: TCR

[SPT3TMS-12320 **]**

Example: [SPT3TMS-12421 **]**

Class PhysicalConsist

{
"structs": [
{
"name": "PhysicalConsist",
"info": "Defines an un-splittable part of the train",
"attrs": [
{"intId": 1, "name": "evnIdentifier", "dataType": "string", "info": "Defines the european vehicle number (EVN) 12 digits"},
{"intId": 2, "name": "passengerLoad", "dataType": "uint32", "unit": "percent", "info": "Defines hte passenger load. Use 255 if unknown"},
{"intId": 3, "name": "mass", "dataType": "uint32", "unit": "kg", "exp": 2, "range": "0..150000", "info": "Defines the train mass including load. Use 0 if unknown"},
{"intId": 4, "name": "maxCurrentConsumption", "dataType": "uint32", "unit": "A", "exp": 1, "range": "0..1023", "info": "Defines the maximum current consumption. Use 1023 if unknown"},
{"intId": 5, "name": "physicalVehicles", "composition": "PhysicalVehicle", "multiplicity": "0..255", "ordered":

"byIndex", "info": "Defines the physical vehicles part of the physical consist. empty=unknown"}
]
}]
} **[**SPT2TS-125698 **]**

10.8.5 Class: PhysicalVehicle

Description:

This class describes the Physical Vehicle of an identified Physical Train Unit provided by a Train Unit Report. **[**SPT3TMS-9762 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
evnIdentifier	uint32	1	European vehicle number (EVN) [2018/1614/EU] 12 digits, 0: unknown
mass	uint32 (kg)	1	Mass of Physical Train Unit (kg) Source: TCR • Range: 0 ... 150000kg Special values: • 15000.1... MAXNUM - 0.1: spare • MAXNUM: Unknown
vehicleOrientation	VehicleOrientation	1	Orientation of Physical Vehicle Source: TCR
doorDataLeft	DoorData	0..7	Positions and states of left doors of the Physical Vehicle Source: TCR
doorDataRight	DoorData	0..7	Positions and states of right doors of the Physical Vehicle Source: TCR
passengerLoad	uint32 (%)	1	Percentage of load in terms of passengers
pantographsStates	PantographState	0..7	Pantographs states

[SPT3TMS-12321 **]**

Example: [SPT3TMS-12457 **]**

Class PhysicalVehicle

```
{
"structs": [
{
"name": "PhysicalVehicle",
"info": "Defines a single carrige inside of a physical consist.",
"attrs": [
```

{"intId": 1, "name": "evnIdentifier", "dataType": "string", "info": "European vehicle number (EVN) [2018/1614/EU] 12 digits"},
{"intId": 2, "name": "mass", "dataType": "uint32", "unit": "kg", "exp": 2, "range": "0..150000", "info": "2047: unknown"},
{"intId": 3, "name": "vehicleOrientation", "enumType": "VehicleOrientation", "info": "Defines the orientation of the vehicle with respect to the train unit"},
{"intId": 4, "name": "doorDataLeft", "composition": "DoorData", "multiplicity": "0..7", "ordered": "byIndex", "info": "Defines the left door data. Left is with respect of normal orientation"},
{"intId": 5, "name": "doorDataRight", "composition": "DoorData", "multiplicity": "0..7", "ordered": "byIndex", "info": "Defines the right door data. Right is with respect of normal orientation"},
{"intId": 6, "name": "passengerLoad", "dataType": "uint32", "unit": "percent", "info": "255=unknown"},
{"intId": 7, "name": "pantographsStates", "enumType": "PantographState", "multiplicity": "0..7", "ordered": "byIndex", "info": "Defines the state of the pantograph"}
]
}
],
"enums": [
{
"name": "DoorState",
"enumLiterals": [
{"intId": 0, "name": "doorStateUnknown"},
{"intId": 1, "name": "doorStateBlocked", "info": "defect"},
{"intId": 2, "name": "doorStateAvailable"}
]
},
{
"name": "PantographState",
"enumLiterals": [
{"intId": 0, "name": "pantographBlocked", "info": "can not be used, defect"},
{"intId": 1, "name": "pantographLowered"},
{"intId": 2, "name": "pantographRaised"},
{"intId": 3, "name": "pantographStateUnknown"}
]
},
{
"name": "DoorSide",
"enumLiterals": [
{"intId": 0, "name": "doorSideLeft"},
{"intId": 1, "name": "doorSideRight"},
{"intId": 2, "name": "doorSideBoth"},
{"intId": 3, "name": "doorNoneSide"},
{"intId": 4, "name": "doorSideUnknown"}
]
},
{
"name": "ActiveState",
"enumLiterals": [
{"intId": 0, "name": "activeState"},
{"intId": 1, "name": "inactiveState"},
{"intId": 2, "name": "undefinedState"}
]
},
{
"name": "VehicleOrientation",
"enumLiterals": [
{"intId": 0, "name": "orientationUknown"},
{"intId": 1, "name": "orientationSameAsTrainUnit"},
{"intId": 2, "name": "orientationInverseToTrainUnit"}
]

}
]
} [SPT2TS-125699]

10.8.5.1 Enum: PantographState

Description:

This enum describes possible pantograph states. **[**SPT3TMS-14201 **]**

Attributes:

Name:	Description:
pantographBlocked	Pantograph blocked
pantographLowered	Pantograph lowered
pantographRaised	Pantograph raised
pantographStateUnknown	Pantograph state unknown

[SPT3TMS-14203 **]**

10.8.5.2 Enum: VehicleOrientation

Description:

This enum describes possible vehicle orientations. **[**SPT3TMS-12609 **]**

Attributes:

Name:	Description:
orientationUknown	Orientation Unknown
orientationSameAsTrainUnit	Orientation same as Train Unit
orientationInverseToTrainUnit	Orientation inverse to Train Unit

[SPT3TMS-12611 **]**

10.8.6 Class: EventHandling

Description:

This class describes the event handling. **[**SPT3TMS-14209 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
event	MovementEvent	1	event
passStopDepart	MovementEventAction	1	Movement event action

[SPT3TMS-14204 **]**

10.8.7 Enum: MovementEventAction

Description:

This enum describes the movement event action. **[**SPT3TMS-14205 **]**

Attributes:

Name:	Description:
passedEvent	Passed event
stoppedAccurateAtEvent	Stopped accurate at event
stoppedWithOvershootAtEvent	Stopped with overshoot at event
stoppedWithUndershootAtEvent	Stopped with undershoot at event
departedFromEvent	Departed from event

[SPT3TMS-15506 **]**

10.8.8 Class: EventEstimation

Description:

This class describes the Event estimation. **[**SPT3TMS-14216 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
event	MovementEvent	1	Event
arrival	timestamp	1	Arrival

[SPT3TMS-14215 **]**

10.8.8.1 Enum: AtoState

Description:

This enum describes possible ATO states: **[**SPT3TMS-12354 **]**

Attributes:

Name:	Description:
Unknown	Unknown
CO	Configuration
NA	Not Available
AV	Available
RE	Ready
EG	Engaged
DE	Disengaged
FA	Failure

[SPT3TMS-12353 **]**

10.9 Train Unit Report Ato Capabilities

10.9.1 Class: TrainUnitReportAtoCapabilities

Description: [SPT3TMS-12325 **]**

This class describes an identified Physical Train Unit provided by ATO Trackside, based on ATO Train Capability Report (TCR). **[**SPT3TMS-12328 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
issuedAt	timestamp	1	Time issued at
operationalPlanMovement	OperationalPlanMovement	0..1	if the new train was created in TrafficCS it does not have an operationalPlan (Movement) yet
trainUnitIdentifier	PhysicalTrainUnitIdentifier	1	Train unit identifier
mass	uint32 (kg)	1	Mass in kg
outsideTemperature	int32 (^{o}C degrees)	1	Outside temperature in Celsius degrees), 127 if unknown
etcsData	EtcsDataAto	0..1	not provided if invalid
passengerLoad	uint32 (%)	1	Passenger load in percent
lastPassengerExchangeTime	uint32 (s)	1	Last passenger exchange time in second
elapsedStartPassengerExchange	uint32 (s)	1	Time elapsed between standstill and the doors are released for operation or opening; 4096: unknown
additionalTrainData	AdditionalTrainData	0..1	Additional train data
physicalConsists	PhysicalConsist	0..*	0-unknown

[SPT3TMS-12322 **]**

Example: [SPT3TMS-12422 **]**

Class TrainUnitCapabilitiesReport and its constituents

{
"structs": [
{
"name": "TrainUnitReportAtoCapabilities",
"info": "Defines train capabilities reported by ATO from the train.",
"attrs": [
{"intId": 1, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 2, "name": "operationalPlanMovement", "reference": "OperationalPlanMovement", "multiplicity":

"0..1", "info": "Refers to the operational plan movement. If the new train was created in TrafficCS it does not have an operationalPlan (Movement) yet"},
{"intId": 3, "name": "trainUnitIdentifier", "composition": "PhysicalTrainUnitIdentifier"},
{"intId": 4, "name": "mass", "dataType": "uint32", "unit": "kg", "exp": 2, "info": "train mass including load, 0 if unknown"},
{"intId": 5, "name": "outsideTemperature", "dataType": "int32", "unit": "degree", "info": "Defines the outside temperature. Use 127 if unknown"},
{"intId": 6, "name": "etcsData", "composition": "EtcsDataAto", "multiplicity": "0..1", "info": "Defines the valid ETCS Data for the train. Not provided if invalid"},
{"intId": 7, "name": "passengerLoad", "dataType": "uint32", "unit": "percent", "range": "0..255", "info": "255 unknown"},
{"intId": 8, "name": "lastPassengerExchangeTime", "dataType": "uint32", "unit": "s", "info": "65535: unknown"},
{"intId": 9, "name": "elapsedStartPassengerExchange", "dataType": "uint32", "unit": "s",
"info": "Defines the time elapsed between standstill and the doors are released for operation or opening; 4096: unknown"},
{"intId": 10, "name": "additionalTrainData", "composition": "AdditionalTrainData", "multiplicity": "0..1", "info": "Defines additional train data as part of ATO capabilities report"},
{"intId": 11, "name": "physicalConsists", "composition": "PhysicalConsist", "multiplicity": "0..255", "ordered": "byIndex", "info": "Defines the list of physical consists. Empty if unknown"}
]
}]
} **[**SPT2TS-125671 **]**

10.9.2 Class: EtcsDataAto

Description:

This class describes the Etcs data Ato. **[**SPT3TMS-14010 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
trainLength	uint32 (m)	1	Train length in meters
maximumSpeed	uint32 (kmh)	1	Maximum speed in km/h
cantDeficiencyCategories	infra.CantDeficiencies	1	Cant deficiency
otherInternationalCategory	infra.OtherSpecificStaticSpeedProfileCategory	0..15	Other international category
axleLoadCategory	infra.LoadCapabilityLineCategories	1	Axle load category in kg
nominalRotatingMass	uint32 (kg)	1	Nominal rotating mass in kg
brakePercentage	uint32 (%)	1	Brake percentage in percent
preconfiguredBrakingModel	uint32	1	Pre-configured braking model
brakePosition	BrakePosition	1	Brake position

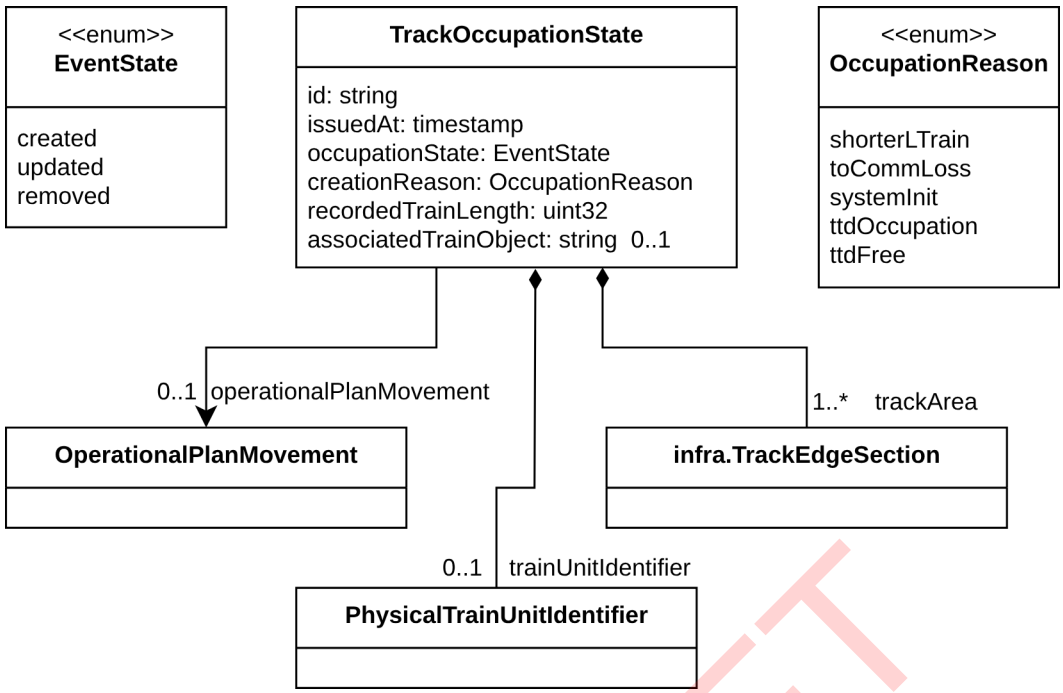
[SPT3TMS-14012 **]**

Example: [SPT3TMS-14224 **]**

{
"structs": [
{
"name": "EtcsDataAto",
"info": "Defines static train capability data reported by ATO",
"attrs": [
{"intId": 1, "name": "trainLength", "dataType": "uint32", "unit": "m", "info": "Defines the train length, 0 if unknown", "ontology": {"subPropertyOf": "http://data.europa.eu/949/length"}},
{"intId": 2, "name": "maxSpeed", "dataType": "uint32", "unit": "km/h", "info": "Defines the maximum applicable speed, use 0 if unknown"},
{"intId": 3, "name": "cantDeficiencyCategory", "enumType": "infra.CantDeficiencies", "info": "Defines the cant deficiency category of the train"},
{"intId": 4, "name": "otherInternationalCategory", "enumType": "infra.OtherSpecificStaticSpeedProfileCategory", "info": "Defines other international static speed profile category ; represents NC_DIFF as per Subset 026-7"},
{"intId": 5, "name": "axleLoadCategory", "enumType": "infra.LoadCapabilityLineCategories"},
{"intId": 6, "name": "nominalRotatingMass", "dataType": "uint32", "unit": "kg", "info": "relates to total train weight. 16 if unknown"},
{"intId": 7, "name": "brakePercentage", "dataType": "uint32", "unit": "percent", "info": "Defines the brake

percentage. 255 if unknown"},
{"intId": 8, "name": "preconfiguredBrakingModel", "dataType": "uint32", "info": "Defines preconfigured braking model for the train. Use 255 if unknown"},
{"intId": 9, "name": "brakePosition", "enumType": "BrakePosition", "info": "Defines the active brake position for the train"}
]
}
]
} **[**SPT2TS-125694 **]**

10.10 Track Occupation State



[SPT2TS-126814]

Figure 24 : Class diagram for Track Occupation State

Maintenance activity: change diagram with that updated by TCCS

10.10.1 Class: UnresolvedTrackboundObject

Description:

This class describes the state (occupied or not) of a track area on the railway network. This information is sent by CCS to TMS when a track occupation is detected.

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	Unique identifier of the Unresolved Trackbound Object
utoExtent	infra.NetAreaReference	1..*	Extension of the Unresolved Trackbound Object
	string	1	

Name:	Type:	Multiplicity:	Description:
associatedTrainObject			Identifier of the Train Object associated to the Unresolved Trackbound Object
recordedTrainLength	uint32 (m)	1	Recorded train length in meters
utoCreationReason	UtoCreationReason	1	Creation reason

Example: [SPT3TMS-12433 **]**

{
"structs": [
{
"name": "UnresolvedTrackboundObject",
"info": "Defines occupation state of a part of the infrastructure graph, sent by low-level system on each track occupation registered, contains CMD.UnresolvedTrackboundObject",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "Id of Unresolved Trackbound Object, used for referencing"},
{"intId": 2, "name": "utoExtent", "composition": "infra.NetAreaReference", "info": "Extent of Unresolved Trackbound Object"},
{"intId": 3, "name": "associatedTrainObject", "dataType": "string", "multiplicity": "0..1", "info": "Reference to Train Object associated to Unresolved Trackbound Object"},
{"intId": 4, "name": "recordedTrainLength", "dataType": "uint32", "unit": "m", "info": "Defines the recorded train length. Use 0 if undefined. Trainlength of the associated Train Object.", "ontology": {"subPropertyOf": "http://data.europa.eu/949/length"}},
{"intId": 5, "name": "utoCreationReason", "enumType": "UtoCreationReason", "info": "Reason for creation of Unresolved Trackbound Object given by ETPS"}
]
}
]} **[**SPT2TS-125672 **]**

10.10.1.1 Enum: UtoCreationReason

Description:

This enum describes possible causes of a trackbound object occupation. **[**SPT3TMS-14230 **]**

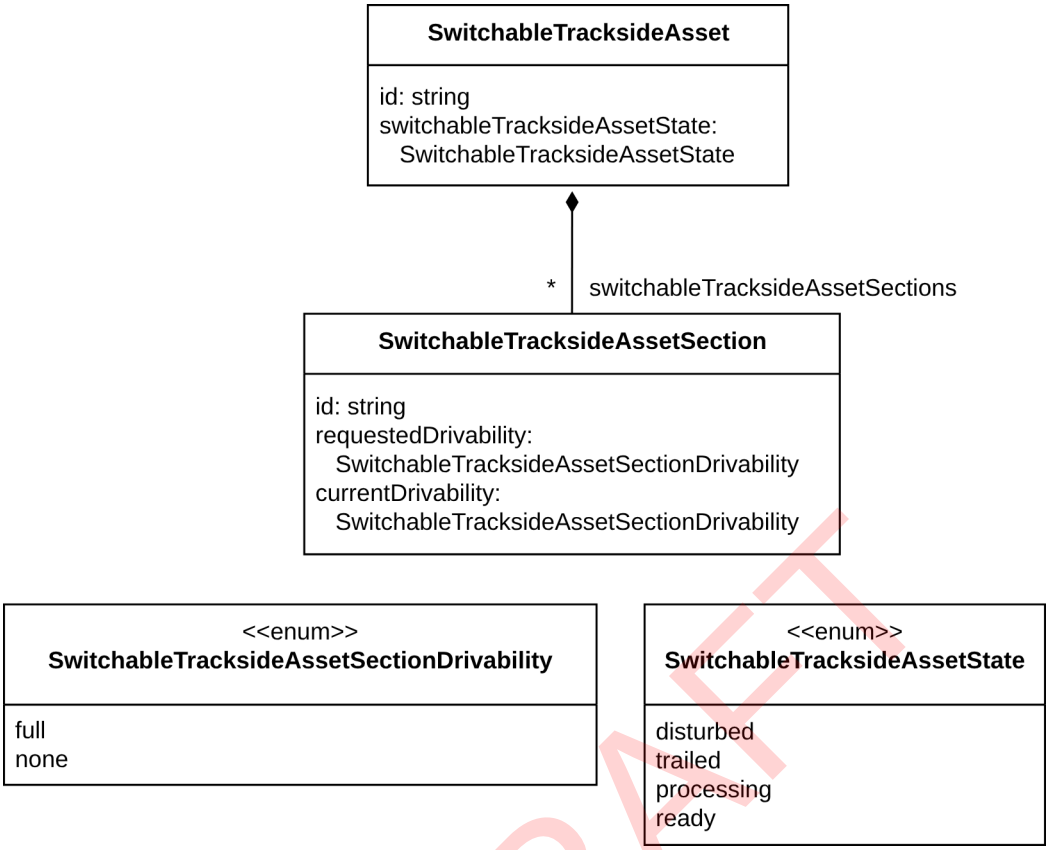
Attributes:

Name:	Description:
ttdFree	ttdFree
systemInit	systemInit

Name:	Description:
shorterLTrain	Shorter L-Train
toCommLoss	ToCommLoss
ttdOccupation	ttdOccupation

[SPT3TMS-14232]

10.11 Switchable trackside asset state



[SPT2TS-126816]

Figure 25 : Class diagram for excludedSwitchableTracksideAssetsForFlankProtection State

10.11.1 Class: SwitchableTracksideAssetState

Description:

This class describes the states of a specific asset which can assume different positions, either current or requested. **[**SPT3TMS-6648 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	Reference to the trackside asset
switchableTracksideAssetState	SwitchableTracksideAssetStateValue	1	State of the switchable trackside asset
switchableTracksideAssetSectionsState	SwitchableTracksideAssetSectionState	1	State of the switchable trackside asset section

[SPT3TMS-12390 **]**

Example: [SPT3TMS-12441 **]**

Class Switchable Trackside Asset State

{
"structs": [
{
"name": "SwitchableTracksideAssetState",
"info": "represents available, requested and current states of the asset",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "sameKeyAs": "tp.SwitchableTracksideAsset",
"info": "Reference to Switchable Trackside Asset"},
{"intId": 2, "name": "switchableTracksideAssetState", "enumType": "SwitchableTracksideAssetStateValue",
"info": "Overall state of Switchable Trackside Asset"},
{"intId": 3, "name": "switchableTracksideAssetSectionState", "composition":
"SwitchableTracksideAssetSectionState", "multiplicity": "*", "info": "Set of Switchable Trackside Asset Sections States that are part of the Switchable Trackside Asset"}
]
}]}

[SPT2TS-125700 **]**

10.11.2 Class: SwitchableTracksideAssetSectionState

Description:

This enum describes the possible drivabilities along the switchable trackside asset section.
[SPT3TMS-14233 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	Id of the switchable trackside asset section
requestedDrivability	SwitchableTracksideAssetSectionDrivability	1	requested driveability of the switchable trackside asset section
currentDrivability	SwitchableTracksideAssetSectionDrivability	1	current driveability of the switchable trackside asset section

[SPT3TMS-14235 **]**

Maintenance activity: add here the link to OPP .json description of this class

10.11.2.1 Enum: SwitchableTracksideAssetStateValue

Description:

This enum describes the possible values for switchable trackside asset states. **[**SPT3TMS-12445 **]**

Attributes:

Name:	Description:
disturbed	Underlying Switchable Trackside Asset State reported as "disturbed". *Details may need to be adjusted once established on SCI-CMD.*
trailed	Underlying Switchable Trackside Asset State reported as "trailed". *Details may need to be adjusted once established on SCI-CMD.*
processing	Switchable Trackside Asset was requested by PES to change and is currently not in any end position (e.g. point is turning, level crossing is opening/closing). *Details may need to be adjusted once established on SCI-CMD.*
ready	Switchable Trackside Asset finished its movement and is in its final position. *Details may need to be adjusted once established on SCI-CMD.*

[SPT3TMS-12447 **]**

Maintenance activity: add here the link to OPP .json description of this class

10.11.2.2 Enum: SwitchableTracksideAssetSectionStateDrivability

Description:

This class describes the possible drivability states of the trackside asset section.
[SPT3TMS-12448 **]**

Attributes:

Name:	Description:
full	*Details may need to be adjusted once established on SCI-CMD.*
none	*Details may need to be adjusted once established on SCI-CMD.*

[SPT3TMS-12450 **]**

Maintenance activity: add here the link to OPP .json description of this class

10.12 Restriction Area State

10.12.1 Class: RestrictionAreaState

Description:

This class describes the Restriction Area State. **[**SPT3TMS-14240 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	ID of restrictionArea generated by interlocking
issuedAt	timestamp	1	Time issued at
restrictionEvent	RestrictionEvent	0..1	Is provided if planned one
restrictionState	RestrictionState	1	Restriction State
restrictionArea	RestrictionArea	0..1	Provided if created by PES or implemented differently than planned

[SPT3TMS-14243 **]**

Example: [SPT3TMS-14245 **]**

{
"structs": [
{
"name": "RestrictionAreaState",
"info": "Defines if the restriction is activated(created)/removed",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "id of restrictionArea generated by interlocking"},
{"intId": 2, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 3, "name": "restrictionEvent", "reference": "RestrictionEvent", "multiplicity": "0..1", "info": "refers to planned restriction event inclusive the operational plan restriction"},
{"intId": 4, "name": "restrictionState", "enumType": "RestrictionState", "info": "Defines if the area was created or removed"},
{"intId": 5, "name": "restrictionArea", "composition": "RestrictionArea", "multiplicity": "0..1", "info": "Defines the restriction area associated to a restriction area state. Provided if created by PE or implemented differently than planned"}
]
}],
"enums": [
{
"name": "RestrictionState",
"enumLiterals": [
{"intId": 0, "name": "created"},
{"intId": 1, "name": "removed"}
]

}
]
} [SPT2TS-127062]

10.12.1.1 Enum: RestrictionState

Description:

This enum describes possible restriction states. [SPT3TMS-14248]

Attributes:

Name:	Description:
created	created
removed	removed

[SPT3TMS-14251]

10.13 Warning Area State

10.13.1 Class: WarningAreaState

Description:

This class describes the Warning Area State. **[**SPT3TMS-14239 **]**

Attributes:

Name:	Type:	Multiplicity:	Description:
id	string	1	WarningArea-id generated by interlocking
issuedAt	timestamp	1	Time issued at
warningMeasureEvent	WarningMeasureEvent	0..1	Warning Measure Event
warningState	EventState	1	Defines the action performed on the the warning area
warningArea	WarningArea	0..1	Provided if differs from warningMeasureEvent.warningArea or unplanned

[SPT3TMS-14238 **]**

Example: [SPT3TMS-14244 **]**

{
"structs": [
{
"name": "WarningAreaState",
"info": "Defines the state of a WarningMeasureEvent (created/updated/removed).",
"attrs": [
{"intId": 1, "name": "id", "dataType": "string", "key": "global", "info": "warningArea-id generated by interlocking"},
{"intId": 2, "name": "issuedAt", "dataType": "timestamp", "info": "Defines the issue-point in time with microsecond-resolution builds a version-id for acknowledgements and validity-estimations, in UTC"},
{"intId": 3, "name": "warningMeasureEvent", "reference": "WarningMeasureEvent", "multiplicity": "0..1", "info": "refers to planned warning measure event (if planned)"},
{"intId": 4, "name": "warningState", "enumType": "EventState", "info": "Defines if the warning area was established or removed"},
{"intId": 5, "name": "warningArea", "composition": "WarningArea", "multiplicity": "0..1", "info": "Defines the warning area associated with the warning area state. Provided if differs from warningMeasureEvent.warningArea or unplanned"}
]
}
]
}

[SPT2TS-127063 **]**

10.13.1.1 Enum: EventState

Description:

This enum describes possible event states. **[**SPT3TMS-16977 **]**

Attributes:

Name:	Description:
created	Created
updated	Updated
removed	Removed

[SPT3TMS-16980 **]**

11 Cross-cutting issues and Non-Functional Requirements

This chapter describes aspects of TMS <> CCS that address both cross-cutting issues as well as non-functional requirements.

11.1 Versioning of Messages

As the traffic situation may change during execution because of conditions in the physical world (e.g. train delays, infrastructure element malfunctions), the Operational Plans need to be adapted regularly. By receiving feedback from CCS sub-systems via the upstream, TMS knows about the actual operational situation and can determine the deviation from the Operational Plan. Based on that, TMS (or responsible staff) might decide to re-optimise the schedule (e.g. change the order of trains) and to elaborate a new version of the Operational Plan. **[**SPT3TMS-9754 **]**

11.1.1 Versioning of Downstream Messages

Version management is to be carried out by TMS. Therefore, a new version of an Operational Plan will be generated by TMS and sent to Plan Execution System and ATO Trackside via TMS <> CCS. On the interface TMS <> CCS it shall be assured that all downstream messages coming from TMS shall be equipped with a unique ID of the Operational Plan as well as a timestamp (issuedAt) of that Operational Plan. **[**SPT3TMS-9755 **]**

11.1.2 Versioning of Upstream Messages

All upstream messages provide a reference to the corresponding Operational Plan, if applicable and available. If this reference can be stated by the sending CCS sub-system, it is required to enrich this message with
a unique ID, as well as a timestamp (issuedAt) of that data object. **[**SPT3TMS-9748 **]**

11.2 Integration of Map Data in Messages on TMS <> CCS

According to what described into the System Architecture Description document, it is assumed that the distribution and the publication of topological map data are centralized and provided by Asset Management System (or another dedicated system) to all those railway systems which either fully or partially rely on it. **[**SPT3TMS-9750 **]**

The topological data model is usually huge and complex, and therefore needs a validation performed by every system which uses it before being applied all over the system. **[**SPT3TMS-16113 **]**

SPT3TMS-16114 - Storing Map Data
The CMS and TMS shall be able to store more than a Map Data model at the same time

This is necessary to ensure a smooth switch from a configuration to another one. **[**SPT3TMS-16120 **]**

SPT3TMS-16115 - Map Data validity period
All Map Data versions stored in CMS and CMS shall have a validity period AND all validity periods shall be fully disjointed one another.

Only a map data model can be active at a given time. **[**SPT3TMS-16118 **]**

SPT3TMS-16119 - Receive Map Data
CMS and TMS shall preferrably rely on the Integration Layer to receive a new configuration of Map Data.

This seems reasonable, in continuity with the achievements of previous initiatives (Shift2Rail), but it is subject to approval by the domain in charge of specifying the Asset Management System, which shoudl publish this infoirmation. **[**SPT3TMS-16528 **]**

SPT3TMS-16117 - Receive Map Data
CMS and TMS shall be able to read a new configuration of Map Data by an external centralized repository (e.g. Digital Register)

To keep both possibilities and be more flexible. **[**SPT3TMS-16124 **]**

SPT3TMS-16125 - Map Data consistency check
CMS and TMS shall pospone the application in service of a new Map Data until all internal checks have been done ensuring consistency and correctness.

This is necessary to ensure a smooth passage to the new configuration without impact neither on TMS/CMS nor on neighbouring systems. This permits all necessary checks to be performed separately also into an off-line environment **[**SPT3TMS-16123 **]**

SPT3TMS-16121 - Map Data Protocol
CMS and CMS shall support OPC-UA protocol

OPC-UA is the protocol chosen to transfer the Map Data model **[**SPT3TMS-16122 **]**

11.3 Timing of Messages

Upstream messages (generated by TCS for TMS) keep TMS updated about the current field situation, enabling a proper reaction and the correct execution of TMS functions. This implies that they have to be sent to TMS a conveniently in advance for TMS for being processed in duly time.

These messages deal with different information and their frequency varies from one to another; therefore, also their timing will be different. This characteristic greatly varies according to several parameters which qualify the controlled area:

- the more trains are present inside the controlled area, the more frequent these messages will be generated.
- A wide controlled areas presumibly shall contain more trains running at a given time, and then more messages will be generated
- a "dense" area, with a high number of trains, is likely to have more rescheduling needs than an area wheer less trains run, which means more messages related to the result of the operational plan and the operating state.

Downstream messages (generated by TMS for TCS) keep TCS updated about the current Operational Plan, and the same considerations done for upstream messages apply. The more trains are present inside the controlled area, the more movements have to be generated and updated, because the likelyhood of deviations and subsequent reschedule increases.

A precise quantitative evaluation shall be possible when some trustable proofs of concept are executed on some significant contexts of application. Here, a theoretical estimation is given as a reference, with some proposed assumptions based on a number of running trains. More scenarios can be of course defined, to perform tests in different conditions and achieve a better estimation in terms of performances to be achieved.

As a reference scenario for movements we might assume an average speed of 120km/h for a train along a line, ignoring stops and assuming that the motion law is rectilinear and uniform, then the train travels about 33m/s. in a traditional system, equipped with on axle counters or audiofrequency track circuits. Assuming for a track circuit an average length of 1350m, it results that every about 40sec. the interlocking gives a change of track circuit to PES module which sends a status change to TMS. Let's assume that the only cause which generates a status change is due to this

In addition to this, taking an ETCS system into account, it is reasonable to assume that every 5 sec an update of the train position is sent by TPS to PES and from PES to TMS. Combining the two events we have a staus message of about 13,5 status messages every minute, i.e. a status message from PES to TMS every 4.3sec. Assuming that for a big Regional Centre there are in the peak hour a number of 50 trains running at the same time, the resulting frequency of the status reports is a message every (4.3/50)*1000=86msec

As it regards the opposite direction, downwards from TMS to TCS, an Operational Plan Operational Movement is sent for a train before it's starting datetime. It is resent everytime a deviation on the Operational plan implies a change in its scheduled trip. In an ideal world no deviations might occur, however in a real world this possibility must be foreseen. It is quite impossible to estimate the frequency of deviations, as the depend from a wide number of parameters. Again, in order to define a reference scenario to be used as a starting point for next considerations, we might assume that every 25 received status messages, one of them notifies a deviation (deviations are then generated in 4% of cases) and implies a reschedule of the operational plan which impacts on 25% of the valid Operational Plan Operational Movements. Keeping the assumption of 50 train active and running at the same time on the controlled line, we have that a deviation is notified every about 86*25=2140msec. This forces a re-evaluation of the 50/4=17,5 Operational Plan Operational Movements (25%) which means an Operation Plan Operational Movement message every 122msec.

The evaluation of timing messages related to Operational Plan Operational Restrictions is yet more difficult as they depend on specific situations: however they are the result of human-driven processes and their frequency is reasonably much lower that movements. Therefore, they should have a very minor impact from the point of view of overall system performances and dealing with them here seems not relevant.

11.4 Reliability

TMS <> CCS is a reactive interface that interacts continuously with other components between TMS and the sub-systems of safety logic. Its behaviour is driven by data fed and forwarding between the systems. TMS <> CCS will distribute data back to its systems environment immediately after receiving it.

Data distribution must occur within defined rules. Any lack of reliability in TMS <> CCS carries the risk of dependent systems also failing to work reliably.

Important aspects to be considered by TMS <> CCS:

- Correctness, i.e., messages are performed without any errors, and any output must be correct and capable of being processed by the environment.
- Concurrency (parallel processing), i.e., the ability to process multiple information channels simultaneously. The capability to process concurrently applies to both requests of an entirely independent nature (e.g., operational plan execution response) and to the joint processing of requests (operational plan execution request).
- Resistance to Resilience against:
    - Errors resulting from inconsistent or incomplete input data (via systems or manual input),
    - Defects,
    - Message overload.
- Defined timing (performance) to:
    - Transmit messages in a timely manner and free of avoidable repetitions (repeated reading or writing of the same data in the same service or context),
    - Perform any sending’s of messages with minimal latency and with one second,
    - Provide immediate user feedback (e.g., upon data entry, mouse click),

- Provide minimum latency via system requests/messages (e.g., operational plan execution report),
- Provide simultaneous (synchronised) channels of information to all relevant systems,
- Enable synchronisation between all server instances minimum latency,
- Near-real-time capabilities able to support future requirements from digitisation projects.

- Elasticity, i.e., the ability to adapt to changing loads and thus to process varying data volumes and frequencies (scalability), and to remain responsive at all times without any negative impact on timing / latency.
- Timing behaviour is intended as follows

[SPT3TMS-9746 **]**

SPT3TMS-16158 - Granularity for commercial / operational data
The granularity (minimum resolution) for commercial / operational data is 1 second.

SPT3TMS-16159 - Granularity for calculations
The granularity (minimum resolution) for calculations, logs, interfaces, and timestamps is 1 millisecond.

11.5 Availability

In case of TMS <> CCS downtime, the systems are no longer supplied with data. There is no automatic generation of data. As a consequence, railway operations realistically cease after some time due to the messages not being transmitted by TMS <> CCS. This is hence neither acceptable for systems operating in normal mode nor in maintenance mode.

Availability is significantly improved via timely detection and proactive elimination of errors. To make this possible, dependent systems are equipped with extensive real-time diagnostic and monitoring functionality. For this purpose, potential error cases and effective error handling strategies addressing each of these must be identified in advance and implemented. Other parts of TMS <> CCS not affected by the error must continue to function without restriction.
The architecture and the operating concept shall support updates, including replacements of hardware, and updates of operating systems, middleware, database, and application versions, without any interruption to system availability. Furthermore, operations-driven changes to master data without any changes to data structures or data semantics shall be carried out during operation and outside maintenance windows. **[**SPT3TMS-9747 **]**

SPT3TMS-16160 - Maximum permitted annual downtime
The maximum permitted annual downtime of TMS <> CCS is 10 minutes, which translates into a mandatory minimum annual availability of 99.998%:

SPT3TMS-16161 - TMS <> CCS availability
TMS <> CCS availability shall not be compromised by maintenance windows with service interruption, and no maintenance windows leading to interface downtime are envisaged.

11.6 Maintainability

TMS <> CCS is predicted to remain in active service for 20 years or longer. High maintainability is hence a crucial pre-requisite for its ability to support successful changes

with reasonable effort – especially given the limited availability of experts on this topic.
To ensure TMS <> CCS´s maintainability:

- **Development** adheres to the EU Rail process.
- **Documentation**, detailed specifications for interfaces and messages, adhere to EU Rail stipulations.
- Designs are:

o Modular, with a decomposition into atomic, individually testable units.
o Utilise:

- Generally known and accepted design patterns.
- Tested components (COTS).
- Industry interface standards.

o Capable of supporting fully automated testing.

- The Implementation:

o Uses a widely used programming language.
o Uses open source whenever possible.
A lack of portability, e.g., to newer versions of standard software products used or to different hardware, significantly reduces maintainability. Therefore, whenever possible, standards must be used in design and adhered to during implementation. The use of and adherence to standards must be ensured throughout the development process, and any use of vendor-specific extensions must be kept to a minimum, encapsulated within the software, and must be documented. An impact analysis must be performed and approved as part of the development process before any deviations from standards, especially pertaining to avoidance of standard products, may occur. A list of approved exceptions shall be kept.
Likewise, the chosen architecture influences maintainability: maintainability is improved by clear definition and documentation of services and their interfaces and by describing:

- The services' task and/or purpose.
- Input / output structures.
- Error and exception handling.
- Conditions (pre- / post conditions).
- Versioning concepts used.

Safety-relevant application rules according to EN 50128 (SAR) Functionality may only be used via the interface published by the sub-system. Implementation details of the sub-systems must remain hidden by their interfaces and thus remain changeable or replaceable. Minimal dependency between sub-systems must be aimed for.
Maintainability further requires that organizational changes, or changes to the operating concept may be performed without changing the software (configurability, decoupling of use and operating concept). In terms of availability, diagnostics and monitoring, the software design must ensure decoupling between operational use and operating concepts (locations, workstation design). Maintainability must be ensured over a long period of time

(> 20 years). Lifecycles of any products selected must be analysed, and their longevity proven.
Good maintainability of TMS <> CCS requires, among other things:

- Support for flexible deployment with an identical architecture with the infrastructure options of:

o On-premises.
o Private cloud.
o Public cloud.

- The use of a web client without the need for any local software installations.
- Consideration of virtualized and standardized server hardware for the technical architecture.

[SPT3TMS-16166 **]**

11.7 Safety

The development of TMS <> CCS components typically has minimal safety requirements i.e., only assure the operational plan is fit for purpose to the operate the railway.
TMS <> CCS is intended as a non-safety-relevant interface and is not intended to assume any responsibility for safety as per to EN 50126-1 or EN 50128. No safety targets are expected for TMS <> CCS, and hence no risk minimizing measures are envisaged for it.
This assumption is based on:

- Comparable functionality in currently used systems being implemented without safety responsibility.
- No safety responsibility being assigned to TMS <> CCS within the overall architecture (system of systems).
- No safety requirements existing for any data propagating through TMS <> CCS.

As soon as SAR (safety-relevant application rules according to EN 50128) of the systems with which interfaces are available, the assumptions will need to be checked. TMS <> CCS must not assume any safety responsibility without the SAR for its messages explicitly requiring and permitting the transmission of safety-relevant data. **[**SPT3TMS-9803 **]**

11.8 Security

The CIA protection goals are **C**onfidentiality, **I**ntegrity, **A**vailability, Authenticity, Privacy, Reliability and (non) repudiation.

- Confidentiality: there are no specific confidentiality-related requirements, hence no special measures addressing either the "protection of information behaviour" or the "protection of information content" are required.
- Integrity: unauthorized data manipulation shall be precluded. Unauthorized data manipulation may impair the availability and reliability of TMS <> CCS. Dedicated integrity requirements must therefore address:

o The correctness of data (data integrity).

o Functional correctness of the interface (system integrity).

- Availability: availability-related requirements for availability are described in general terms. Additional considerations from a security perspective are that the system's availability must not be compromised by attacks upon it.
- Authenticity: Special requirements exist regarding:

o Proof of identity of all system partners (partner authenticity).

o Proof that the data received originates from the authenticated instance (data authenticity).

- Privacy: There are no special requirements regarding privacy, and communication processes are not subject to any secrecy requirements. TMS <> CCS is not intended to support anonymous use.
- Accountability: There are no special requirements regarding immutability. Technical traceability of use (e.g., via data logging) must be ensured.
- Non-repudiation: There are no special requirements for non-repudiation. Technical traceability of use (e.g., via data logging) must be ensured.

TMS <> CCS must provide medium levels of protection against intentional violations of security requirements. It shall, however, be accepted that this level of protection will be surmountable to persons with average knowledge of the system applying a medium amount of effort. **[**SPT3TMS-9799 **]**

The following non-functional requirements apply: **[**SPT3TMS-16162 **]**

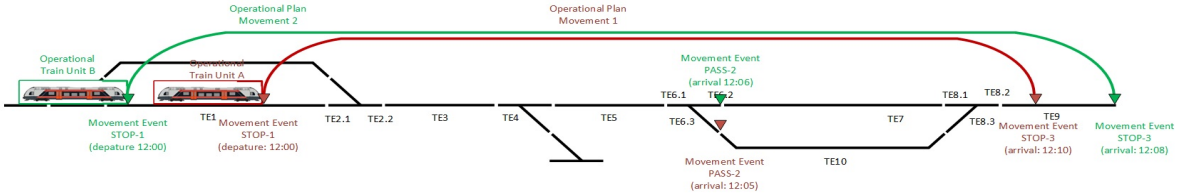
SPT3TMS-16163 - Data Manipulation
Unauthorized data manipulation shall be precluded.

SPT3TMS-16164 - Integrity breach
The design of the interface shall ensure that the system cannot be compromised

SPT3TMS-16165 - Risk Assessment
The TMS-CCS interface shall be part of the system Security Risk Assessment.

12 Appendix

12.1 Example of an Operational Plan Movement in JSON-Format



[SPT3TMS-15492]

Figure 26 : Schematic Operational Plan Movements for two train runs passing

The following example illustrates two Operational Plan Movements for two train runs passing each other. Train Unit A will operate the train run followed by Train Unit B. However, Train Unit B will overpass Train Unit A between TE6.2 and TE8.1. As a result, the order of the Train Units will be changed after that.
[SPT3TMS-15501]

Example OperationalPlanMovement Train Unit A: [SPT3TMS-15499 **]**


```
{
"id": "ID-OPOM-1",
"issuedAt": "2024-02-02T11:00:00.53235",
"configurationDataVersionRef": "2.0",

"includesEndOfJourney": "false",
"operationalTrainUnits": [
{
"physicalTrainUnit": {
"trainRunningNumber": "A",
"additionalTrainRunningNumber": "",
"tafTapTsiTrainID": "",
"leadingOnBoardUnitId": "9920022"
},
"operationalTrainCategory": {
"type": "regional",
"description": "",
"trainUsage": "locomotiveRunningLight"
},
"operationalConsists": [
{
"supportedOnBoardEquipment": {
"type": "Etcs",
"etcsSystemVersions": [
"1.0",
"1.1",
"1.2"
]
}
}
],
"nominalRotatingMass": 10,
"regularBrakeWeightPercentage": 150,
"brakePosition": "passengerTrainInP",
"brakeModelType": "gamma",
```

```
"usedGradeOfAutomation": "GoA2"
}
],
"movementEvents": [
{
"id": "ID-ME-STOP-1",
"plannedType": "stop",
"scheduledType": "stop",
"alignment": "head",

"position": {

"hasTopoCoordinate": {

"onLinearElement": {

"id": "TE_1",

"lengthOfNetLinearElement": "400",

},

"offsetFromOrigin": "100"

}

},
"pathToNextEvent": [
{
"id": "TE2.1",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE2.2",
"lengthOfNetLinearElement": "400"
},
{
"id": "TE3",
"lengthOfNetLinearElement": "1350"
},
{
"id": "TE4",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE5",
"lengthOfNetLinearElement": "1350"
},
{
"id": "TE6.1",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE6.3",
"lengthOfNetLinearElement": "150"
}
],
"stopDescription": {
"doorActivity": {},
"relaxedCoupler": false,
"holdTrain": false,
```

"scheduledDeparture": "2024-03-01T12:00:00",
"scheduledMinDwellTime": 300,
"additionalEventTimes": []
},
"scheduledArrival": "2024-03-01T11:55:00",
"scheduledArrivalWindow": 60
},
{
"id": "ID-ME-PASS-2",
"plannedType": "pass",
"scheduledType": "pass",
"alignment": "head",

"position": {

"hasTopoCoordinate": {

"onLinearElement": {

"id": "TE_6.3",

"lengthOfNetLinearElement": "150"

},

"offsetFromOrigin": "0"

}

},

"pathToNextEvent": [
{
"id": "TE10",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE8.3",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE8.2",
"lengthOfNetLinearElement": "400"
},
{
"id": "TE9",
"lengthOfNetLinearElement": "1350"
}
],
"scheduledArrival": "2024-03-01T12:05:00",
"scheduledArrivalWindow": 60
},
{
"id": "ID-ME-STOP-3",
"plannedType": "stop",
"scheduledType": "stop",
"alignment": "head",

"position": {

"hasTopoCoordinate": {

"onLinearElement": {

"id": "TE_9",

"lengthOfNetLinearElement": "1350"

},

"offsetFromOrigin": "0"

}

},

"stopDescription": {
"doorActivity": {},
"relaxedCoupler": false,
"holdTrain": false,
"scheduledDeparture": "2024-03-01T12:15:00",
"scheduledMinDwellTime": 300,
"additionalEventTimes": []
},
"scheduledArrival": "2024-03-01T12:10:00",
"scheduledArrivalWindow": 60
}
]
}

Example OperationalPlanMovement Train Unit B: [SPT3TMS-15503 **]**


```
{
"id": "ID-OPOM-2",
"issuedAt": "2024-02-02T11:00:00.53235",
"configurationDataVersionRef": "2.0",

"includesEndOfJourney": "true";
"operationalTrainUnits": [
{
"physicalTrainUnit": {
"trainRunningNumber": "B",
"additionalTrainRunningNumber": "",
"tafTapTsiTrainID": "",
"leadingOnBoardUnitId": "9920021"
},
"operationalTrainCategory": {
"type": "regional",
"description": "",
"trainUsage": "locomotiveRunningLight"
},
"operationalConsists": [
{
"supportedOnBoardEquipment": {
"type": "Etcs",
"etcsSystemVersions": [
"1.0",
"1.1",
"1.2"
]
}
}
],
"nominalRotatingMass": 10,
"regularBrakeWeightPercentage": 150,
"brakePosition": "passengerTrainInP",
"brakeModelType": "gamma",
"usedGradeOfAutomation": "GoA2"
}
],
"movementEvents": [
{
"id": "ID-ME-STOP-1",
"plannedType": "stop",
"scheduledType": "stop",
"alignment": "head",

"position": {

"hasTopoCoordinate": {

"onLinearElement": {

"id": "TE_1",

"lengthOfNetLinearElement": "400",

},

"offsetFromOrigin": "50"

}
```

```
},
"pathToNextEvent": [
{
"id": "TE2.1",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE2.2",
"lengthOfNetLinearElement": "400"
},
{
"id": "TE3",
"lengthOfNetLinearElement": "1350"
},
{
"id": "TE4",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE5",
"lengthOfNetLinearElement": "1350"
},
{
"id": "LE6.1",
"lengthOfNetLinearElement": "450"
},
{
"id": "TE6.2",
"lengthOfNetLinearElement": "100"
}
],
"stopDescription": {
"doorActivity": {},
"relaxedCoupler": false,
"holdTrain": false,
"scheduledDeparture": "2024-03-01T12:00:00",
"scheduledMinDwellTime": 300,
"additionalEventTimes": []
},
"scheduledLatestArrival": "2024-03-01T11:55:00",
"scheduledArrivalWindow": 60
},
{
"id": "ID-ME-PASS-2",
"plannedType": "pass",
"scheduledType": "pass",
"alignment": "head",
"position": {
"hasTopoCoordinate": {
"onLinearElement": {
"id": "TE_6",
"lengthOfNetLinearElement": "150"
```

```
},
"offsetFromOrigin": "0"
}
},
"pathToNextEvent": [
{
"id": "TE7",
"lengthOfNetLinearElement": "400"
},
{
"id": "TE8.1",
"lengthOfNetLinearElement": "800"
},
{
"id": "TE8.2",
"lengthOfNetLinearElement": "400"
},
{
"id": "TE9",
"lengthOfNetLinearElement": "1350"
}
],
"scheduledLatestArrival": "2024-03-01T12:06:00",
"scheduledArrivalWindow": 60
},
{
"id": "ID-ME-STOP-3",
"plannedType": "stop",
"scheduledType": "stop",
"alignment": "head",
"position": {
"hasTopoCoordinate": {
"onLinearElement": {
"id": "TE_9",
"lengthOfNetLinearElement": "1350"
},
"offsetFromOrigin": "0"
}
},
"stopDescription": {
"doorActivity": {},
"relaxedCoupler": false,
"holdTrain": false,
"scheduledDeparture": "2024-03-01T12:15:00",
"scheduledMinDwellTime": 300,
"additionalEventTimes": []
},
"scheduledLatestArrival": "2024-03-01T12:10:00",
"scheduledArrivalWindow": 60
```

```
  }
  ]

  }
```

12.2 Open Points

This chapter describes the open points which will be addressed in future releases of this concept.

12.2.1 Prio 1

All priority 1 topics have been managed in 2.0 release

12.2.2 Prio 2

- Challenge return codes of feasibility checks of Operational Plan Execution Response

[SPT3TMS-9801]

12.2.3 Prio 3

- Incorporate versioning of MAP data in all messages

o among others: Map data version in Operational Events

- Challenge whether PES Alarms must be considered in Upstream. Open: Will we get all necessary data from TPS?
- Update Class: StopTimes
- Update Class: StopActivity
- Integrate message descriptions and attributes, as well as all classes of data model to RCA Capella-model as “candidates”
- Analyse if an abstract concept is required, to report differences (e.g. added/removed Operational Events, different referenced track sections) between the Operational Plan and its actual implementation, back to TMS.

o This does not concern differences in planned and implemented times of Operational Events

o Differences may for example occur due to interaction of SubSys Safety Manager or SubSys Workbench with SubSys Safety Logic

- Introduction of underlying design pattern “Published Language (DDD pattern)” of the Operational Plan
- Address further open points in the chapters 3 and 4 (marked in red italic text)
- Review and update consideration of GoA-Level, door-opening behaviour etc. in downstream and upstream, if required
- Update Figure 14to show Operational Segments
- Analyse if reference TRID of TSI TAF/TAP shall be integrated in Class: Movement
- For attribute ‘position’ in 10.2.10: Would it be useful to provide a quality indicator for ATO which describes how well this position shall be reached by a TU? This is currently not available in SS126.
- Validate use cases of TCR, check necessity of attributes and impact on TMS <> CCS
- Can we replace DirectedTrackPath with directed LCTA? See 10.9.6.
- Why do we need the system version in 10.2.8? Isn’t this rather a question of “route compatibility check”?

[SPT3TMS-9802]

12.2.4 Collected points for aligning the Train Unit Report with TPS

- Comment by Frank Skowron: What is “SAFE_AGGREGATED” in 10.6.1for? The only information which is deriveable is that the rear end was extended to the next TVPS section boarder and is very likely shorter BUT knowbody knows if it is really shorter.
- Attribute trainLength in 10.6.4should be entity 1, not 0..1. But it is probably not mandatory in the TCR.
- Modification of attribute ‘velocity’ in 8.1.3.28.1.3.1: Localization differs between velocity (a vector) and speed (scalar). As understood, velocity is more than speed AND direction (in the sense of forward/backward). Propose to approach e.g. Muthu for that.
- Clarify meaning of attribute ‘vehicleOrientation’ in 10.6.9: It is unclear whether this is to identify

o either that one vehicle of a PTU has the right/wrong orientation or

o the whole PTU has the right/wrong orientation?

With b), also the position of a vehicle in the PTU changes.

- Clarify meaning of attribute ‘lastDoorOpeningTime’ in 10.6.6: It would be interesting to know the use case. Is this information required to be received near-time.

[SPT3TMS-9795 **]**