RCA

Reference CCS Architecture

An initiative of the ERTMS users group and the EULYNX consortium

Concept: Standard Maintenance Interface (SMI)

Preliminary issue

Document id: RCA.Doc.74

Table of contents

List of Figures

List of Tables

Version History

Version	Date	Author	Description
0.1	2022-02-25	Bettina Morman	Initial draft
0.2	2022-05-30	Bettina Morman, Martin Kemkemer, Harish Narayanan, Jens Wieczorek, Johannes Elstner, Saku Salo, Marius Kunde	Version after internal review
0.3	2022-06-28	Bettina Morman, Martin Kemkemer, Harish Narayanan, Jens Wieczorek, Johannes Elstner, Saku Salo, Marius Kunde	Version after preliminary review
1.0	2022-09-08	Bettina Morman, Martin Kemkemer, Harish Narayanan, Saku Salo, Marius Kunde	Version for RCA BL1R0

1. Introduction

1.1 Release information

RCA-Document Number: RCA.Doc.74

Document Name: Concept: Standard Maintenance Interface (SMI)

CENELEC Phase: 1

Version: 1.0

RCA Baseline set: BL1R0

Approval date: 2022-09-30

1.2 Imprint

Publisher:

RCA (an initiative of the ERTMS Users Group and EULYNX Consortium)

Support and Feedback:
For feedback, or if you have trouble accessing the material, please contact rca@eulynx.eu.

1.3 Disclaimer

This issue is a preliminary version of this document. The content of this document reflects the current ongoing specification work of RCA. Formal requirements management and change management will be introduced in future iterations. The content may be unfinished, will likely contain errors and can be changed without prior notice.

1.4 Purpose of this document

This concept document describes the design, principles, and structure of the Standard Maintenance Interface (SMI). The SMI connects the subsystem Device & Configuration Management (DCM) with all RCA subsystems located at trackside, as described in 2.1.

This concept defines the SMI for the connection between the DCM subsystem and trackside RCA subsystems. The concept does not consider SCI-Interfaces to be defined to distribute RCA Configuration Data to on-board systems. Nevertheless, the concept must consider the fact that RCA relevant Configuration Data for on-board systems must be distributed via SMI to trackside RCA subsystems before they can be passed on from there via SCI-Interfaces to the on-board systems.

The evaluation of different approaches to publish RCA Configuration Data to On-Board (PUB-OB) can be found in [10]. The detailed system definition for the selected approach out of [10] along with relevant functional allocations to the RCA subsystems can be found in the [12].

1.5 Maturity and Related Topics

This document is the enhanced publication on this topic from RCA. It depicts the concept of the SMI. The main purpose is to establish the topic further and to continue discussions with stakeholders in the sector.

The concept is work in progress. Where it is already known that a section needs further elaboration, it is highlighted like this: *This section will be further elaborated in future releases.*

1.6 Structure of document

This concept is structured as follows:

- Chapter 1 provides brief introductions to the SMI Concept.
- Chapter 2 provides the scope and purposes of the SMI.
- Chapter 3 explains the interface design and principles of the SMI.
- Chapter 4 explains the detailed structure of the SMI.
- Chapter 5 explains the cross-cutting issue and NFR's.
- Chapter 6 provides the list of open points.
- Chapter 7 is the appendix for SMI concept.

1.7 Related documents

The following related RCA documents provide further information and build on this concept:

[1] RCA System Definition [RCA.Doc.35]
[2] RCA Terms and Abstract Concepts [RCA.Doc.14]
[3] RCA MAP Object Catalogue [RCA.Doc.69]
[4] RCA Solution Concept: MAP [RCA.Doc.54]
[5] System Requirements Specification, Subset-026-7 v3.6.0
[6] Interface definition and specification SMI [Eu.Doc.76]
[7] RCA A.P.M. Business strategy and targets [RCA.Doc.50]
[8] RCA A.P.M. Objectives [RCA.Doc.53]
[9] RCA Capella Model (Architecture) – Capability SysC87 "Activate Map Data"
[10] RCA Digital Map – Evaluation Publish Onboard Map Approaches [RCA.Doc.56]
[11] RCA Methods and Tooling Arch Process
[12] RCA Digital Map System Definition [RCA.Doc.59]
[13] RCA Architecture Poster [RCA.Doc.40]

1.8 Writing conventions

The following writing conventions from [11] apply to this document:

- Abstract concepts are written with the first letters capitalised and with spaces
- Messages are written in title case
- Classes are written in pascal case
- Attributes are written in camel case

2. Standard Maintenance Interface

2.1. Introduction

Standard Maintenance Interface

The Standard Maintenance Interface (SMI) is part of the Reference CCS Architecture (RCA), see Figure 1. It is together with the Standard Authentication/Authorisation Interface (SAI) and the Standard Diagnostics and Monitoring Interface (SDI) one of the so-called ‘cross-cutting’ interfaces of RCA. SMI connects the subsystem DCM with all trackside RCA subsystems. The SMI is a bi-directional interface.

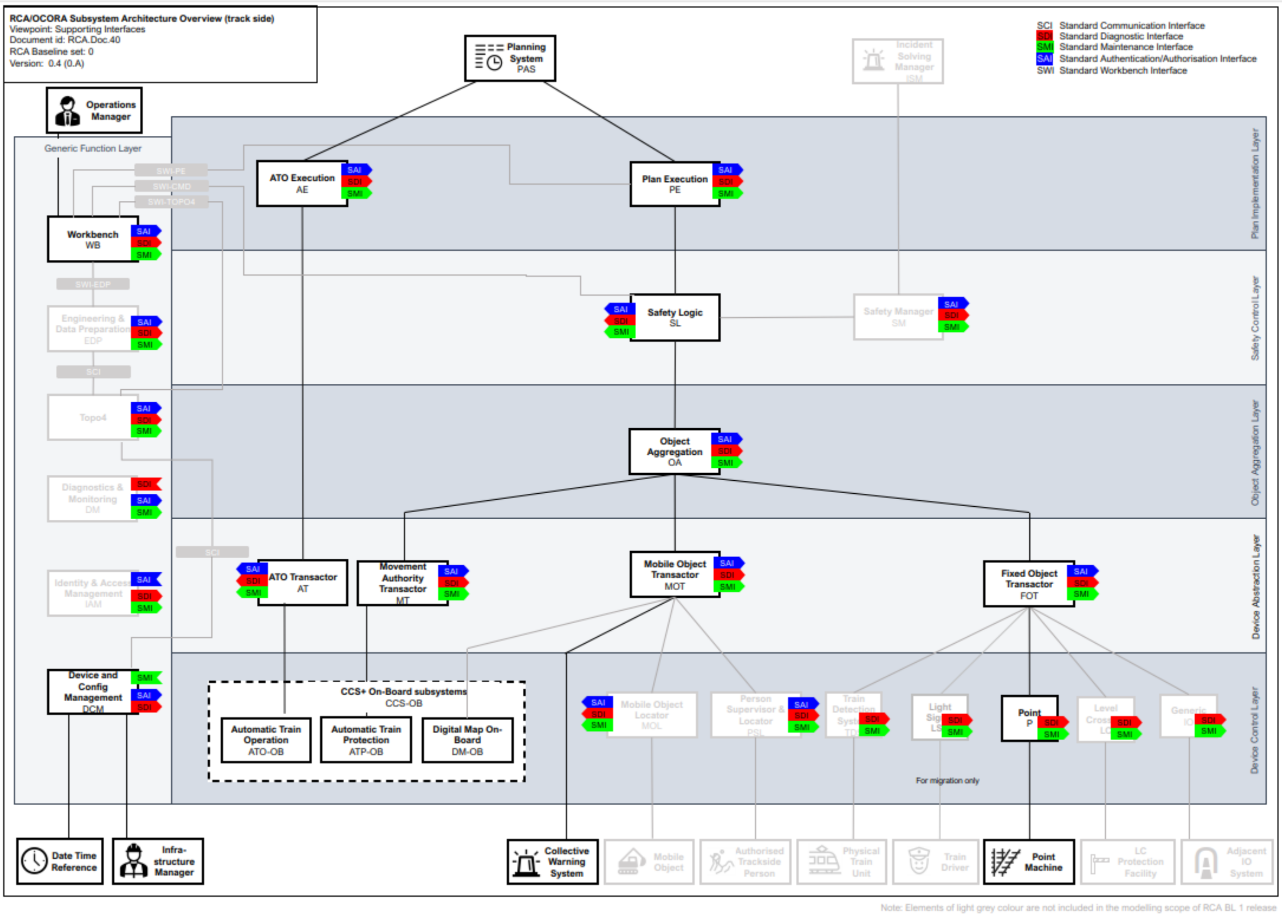

Figure 1: RCA/OCORA Subsystem Architecture Overview (track side), Viewpoint: Supporting Interfaces

The following table shows the different types of the SMI and the connected RCA trackside subsystems. All these types of the SMI shall be covered by one standardised specification of the SMI.

Name	Primary end	Secondary end
SMI-MOT	DCM	Movable Object Transactor (MOT)
SMI-SL	DCM	Safety Logic (SL)
SMI-SM	DCM	Safety Manager (SM)
SMI-PE	DCM	Plan Execution (PE)
SMI-MT	DCM	Movement Authority Transactor (MT)
SMI-OA	DCM	Object Aggregator (OA)

SMI-FOT	DCM	Fixed Object Transactor (FOT)
SMI-AE	DCM	ATO Execution (AE)
SMI-AT	DCM	ATO Transactor (AT)
SMI-TOPO4	DCM	TOPO4
SMI-EDP	DCM	Engineering & Data Preparation (EDP)
SMI-DM	DCM	Diagnostics & Monitoring (DM)
SMI-IAM	DCM	Identity & Access Management (IAM)
SMI-WB	DCM	Workbench (WB)
SMI-MOL	DCM	Mobile Object Locator (MOL)
SMI-PSL	DCM	Person Supervisor & Locator (PSL)
SMI-TDS (EULYNX)	DCM	Train Detection System (TDS)
SMI-LS (EULYNX)	DCM	Light Signal (LS)
SMI-P (EULYNX)	DCM	Point (P)
SMI-LC (EULYNX)	DCM	Level Crossing (LC)
SMI-IO (EULYNX)	DCM	Generic IO (IO)

Table 1: Standard Maintenance Interfaces

Subsystem Device & Configuration Management

At the present time, there is no subsystem definition of the subsystem DCM. However, it can be descripted as follows:

The subsystem DCM is used to register, setup, and configure Devices. This includes distributing and updating the Configuration Data and the software version to the RCA subsystems.

Characteristics:

- *Subsystem DCM might be standardised only partially. While it provides standardised interfaces (SMI) towards Sys RCA, only the core functionality required for RCA subsystems are specified. Its functions will not be fully specified by RCA.*
- *Safety criticality: Subsystem DCM is assumed to be safety critical, see [13]*

It should be noted that the description of the subsystem DCM above is an assumption and therefore, the actual functional and non-functional requirements may differ from this description.

2.2. Categorisation of data

There are different types of data being distributed between the RCA subsystems. On a high level, they can be categorised into Operational Data and Configuration Data (see figure below). Further types of data are present (e.g. diagnostics data, authentication data) but out of scope here because they are not transferred via SMI.

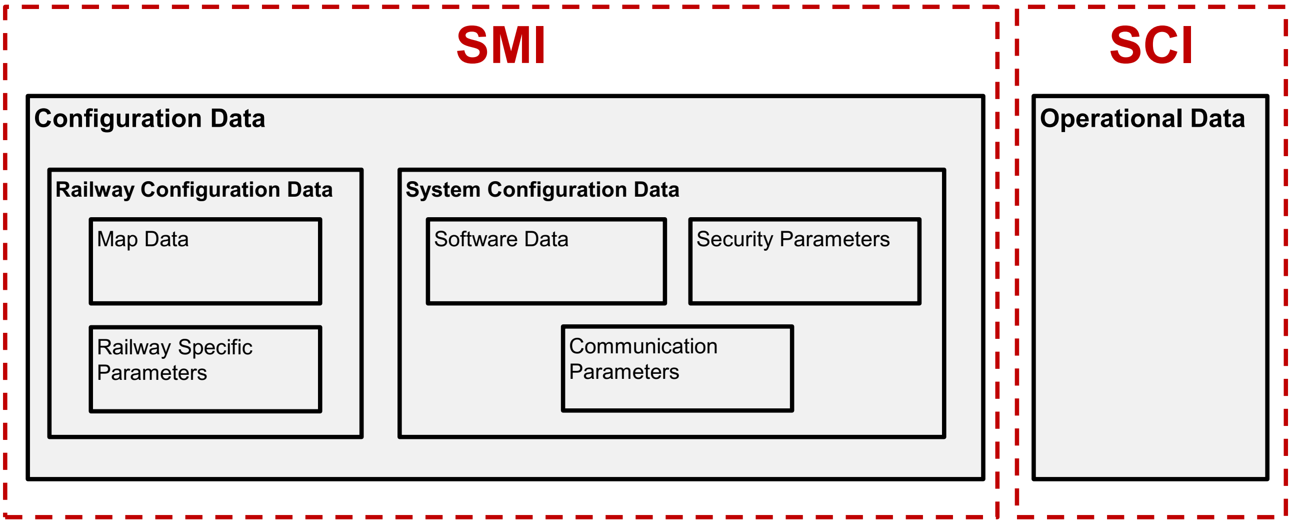

Figure 2: Categorisation of data

2.2.1. Operational Data

Operational Data is the data transferred to/from/between RCA subsystems via the RCA Standard Communication Interfaces (SCI) as well as the interfaces defined by EULYNX and CCS TSI Subsets (ETCS and AoE). It includes runtime data gathered or transferred during railway operation i.e. during the movements of trains. Examples of Operational Data are

- Operational Plan
- Domain Objects, e.g. Movement Permission, DPS
- Train Position Report
- Status Report

2.2.2. Configuration Data

Configuration Data is the data distributed between the RCA subsystems DCM and other RCA subsystems via the SMI providing configurations that concern different functionalities and applications in the RCA subsystems. Whenever possible, the subsystems can generally be updated with new Configuration Data during operation without the need to 'reboot'. This means that the area being updated is usually not available for normal operation in some cases, a maintenance window might be necessary for update reasons, e.g. for Object Controller (OC) configuration. Configuration Data is split into two categories: Railway Configuration Data and System Configuration Data.

Railway Configuration Data

Railway Configuration Data is comprised of Map Data and Railway Specific Parameters describing the characteristics of a railway infrastructure and its operative specifics. These two groups of data are defined as follows (see also RCA.Doc.14 Terms and Abstract Concepts [2]):

a) **Map Data**
Map Data contains a very detailed microscopic digital representation of the railway network that contains all information necessary for planning and performing railway operations, such as infrastructure characteristics, location and details of Field Elements, etc. The Map Data remain unchanged until the next provisioning of Map Data.

b) Railway Specific Parameters

While the configuration of the RCA subsystems shall mainly consist of Map Data it is complemented by additional Railway Specific Parameters. Unless explicitly included in the respective context, the

Railway Specific Parameters are not part of the Map Data. However, the following dependencies should be considered:

- The Railway Specific Parameters will be provided in the same way as Map Data
- A consistent configuration requires at least partial alignment between Map Data and system parameters.

Railway Specific Parameters also consider national and specific operative environments. Thus, they form the specific application of the generic system.

Examples of Railway Specific Parameters are:

- APS: National safety rules, safety patterns for supervision of safety logic, and the configuration of safety checks
- PE: Min/max extent of Movement Permissions, frequency of requesting Movement Permissions
- OC: Drive engine type of the point machine/other track equipment, threshold for turnaround time
- CCS-OB: ATO-TS default contact information, national values

System Configuration Data

System Configuration Data consists of the IT infrastructure data necessary to manage and run software and hardware systems. In addition, it also enables safe and secure communication between systems and its constituents. System Configuration Data is split into three groups of data:

a) Software Data
Software Data includes the application software (SW) and SW configuration that “tailor” a generic SW to a specific application and use. Examples of software data are

- SW version
- HW/SW compatibility matrix
- Operating system configuration (e.g., logging parameter)

b) Security Parameters
Security parameters comprise configurable data that protect critical SW and communication infrastructure from cyber-attacks. Security parameters can be set to specify characteristics of the security design implementation, such as identification and authentication control, use control, system integrity, data confidentiality, and restricted data flow.

Examples of security parameters are authentication certificates and public keys.

c) Communication Parameters
Communication parameters configure basic settings for data transmission between systems, thus specifying the communication method systems exchange information.
Examples of communication parameters are

- Network addresses, bandwidth
- Protocol versions, protocol stack
- System versions required, session timeout values
- Bit rate, parity, number of (stop) bits

2.3. Business targets and objectives

The business targets and objectives from the document [8] that are relevant for the SMI can be found in the appendix (section 7.2).

2.4. System Requirements

The following tables list the system requirements for SMI, derived from the business targets and objectives listed in the annex.

Category	System Requirement
Distribution of Configuration Data	SMI shall provide the DCM subsystem the capability to distribute generic Configuration Data to trackside RCA subsystems.
	SMI shall provide the DCM subsystem the capability to distribute specific Configuration Data to trackside RCA subsystems.
	SMI shall provide the DCM subsystem the capability to distribute Configuration Data to trackside RCA subsystems frequently
	SMI shall provide the DCM subsystem the capability to distribute Configuration Data to trackside RCA subsystems efficiently
	SMI shall provide the DCM subsystem the capability to distribute Configuration Data to trackside RCA subsystems safely and securely
	SMI shall provide the DCM subsystem the capability to distribute Configuration Data to trackside RCA subsystems without affecting system operation
	SMI shall provide the DCM subsystem the capability to distribute Configuration Data to trackside RCA subsystems during runtime
	SMI shall provide the DCM subsystem the capability to be made accessible for external (legacy) systems.
Activation of Configuration Data	SMI shall provide the DCM subsystem the capability to activate generic Configuration Data to trackside RCA subsystems.
	SMI shall provide the DCM subsystem the capability to activate specific Configuration Data to trackside RCA subsystems.
	SMI shall provide the DCM subsystem the capability to activate Configuration Data to trackside RCA subsystems frequently
	SMI shall provide the DCM subsystem the capability to activate Configuration Data to trackside RCA subsystems efficiently
	SMI shall provide the DCM subsystem the capability to activate Configuration Data to trackside RCA subsystems safely
	SMI shall provide the DCM subsystem the capability to activate Configuration Data to trackside RCA subsystems with minimum impact on system operation
	SMI shall provide the DCM subsystem the capability to activate Configuration Data to trackside RCA subsystems during runtime

2.5. Scope

The Standard Maintenance Interface provides an interface to all trackside RCA subsystems which require Configuration Data as a part of their application layer.

The scope of the SMI is the distribution and activation of Configuration Data (as described in chapters 3.1 and 3.2) to all RCA subsystems located trackside. This includes the scope of the SMI defined by EULYNX [6] (SMI-TDS, SMI-LS, SMI-P, SMI-LC, SMI-IO).

Unlike SCI, SMI does not include the feature of distributing RCA Configuration Data to on-board systems. Nevertheless, SMI must consider the fact that RCA relevant Configuration Data for on-board systems must be distributed via SMI to trackside RCA subsystems before they can be passed on from there via SCI-Interfaces to the on-board systems.

2.6. Use Cases

The following exemplary use cases of RCA have been identified as relevant for the SMI:

1. Perform the system configuration and activation during initial setup or start-up of RCA subsystems.
2. Perform the system configuration and activation when introducing a new Object Controller (OC).
3. Perform the configuration update of RCA according to the requirements of the IM for operational behaviour during runtime.
4. Perform the configuration update of RCA according to the requirements of the IM for safety behaviour during runtime.
5. Perform Map Data updates of RCA trackside subsystems and CCS-OB during runtime.
6. Perform configuration of the CCS-OB with IM-specific on-board configuration (e.g. with national values) updated at the IM borders.
7. Perform configuration of the CCS-OB at the AoC borders (e.g. set default ATO-TS / ATO-Transactor to be contacted).

3. Interface design and principles

3.1. Distribution principles for Configuration Data

The distribution principles for Configuration Data are divided into three sections to describe the distribution principles for Map Data, Railway Specific Parameter Data, and System Configuration Data.

3.1.1. Distribution of Map Data

The distribution principles for Map Data are described using a base concept and specific scenarios related to different Map Data distribution use cases.

3.1.1.1. Concept

The conceptual analysis carried out as a part of RCA.Doc.54 [4] provides a basic distribution logic which involves Map Data being distributed (in a synchronised way) by DCM to the RCA subsystems and a corresponding acknowledgment being provided back to DCM by the subsystems. The sequence diagram below briefly shows the Map Data distribution logic.

Note: The illustration below is only preliminary approach and does not show all the potential subsystems requiring Map Data. In addition, this illustration does not portray/propose any kind of unicast, multicast, or point to point distribution principles. The idea here is to only impart the need of Map Data for subsystems and introduced DCM as a Map Data distributor.

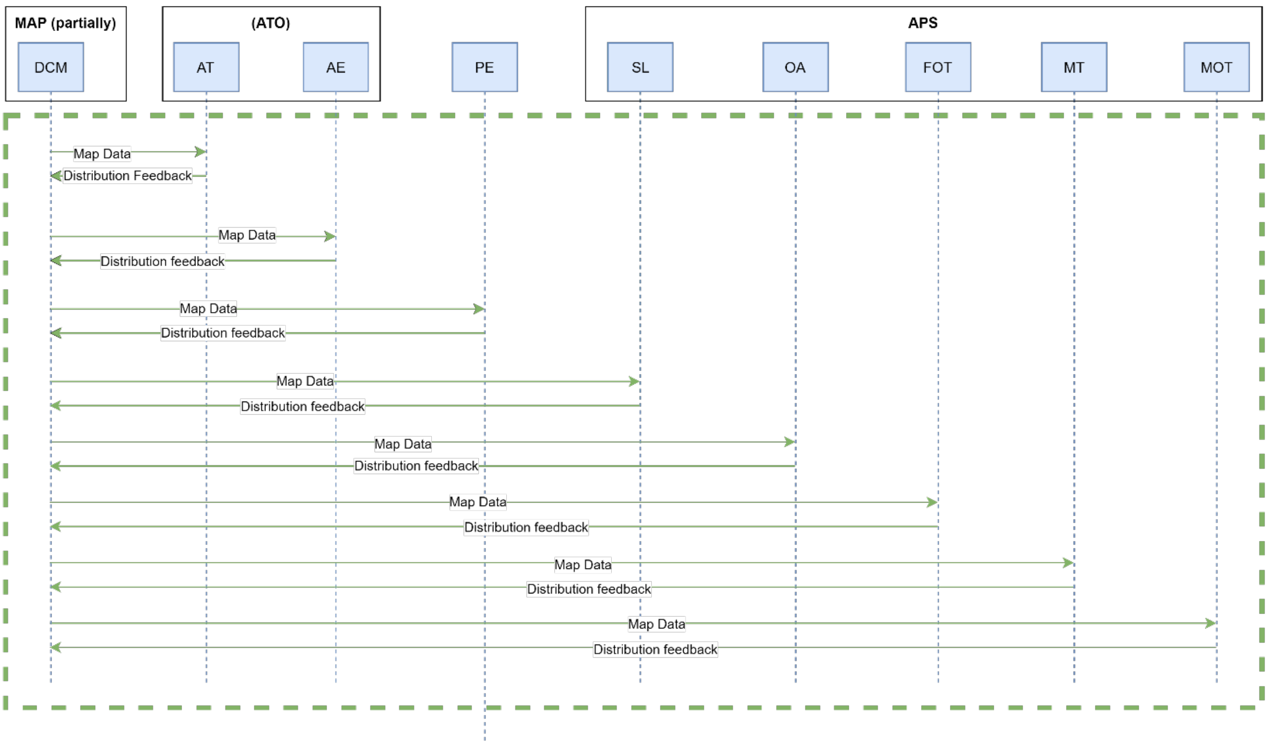

Figure 3: Concept - Distribution of Map Data

For detailed information regarding the Map Data distribution, refer to [4].

3.1.1.2. Scenario: Initial distribution of Map Data to all trackside subsystems

During the initial case of distribution of Map Data (i.e., when the subsystems do not have any Map Data present), every subsystem requests Map Data from subsystem DCM using an initial Map Data request. This initial request indicates information regarding existing Map Data version within the subsystem.

Note: It is known that there is no existing Map Data during initial distribution, but it is done to ensure that we have a generic version check process in DCM (see in section 3.2.4). A solution in such a case could be that the Map Data version information is left out or filled with NULL.

This existing version information from subsystems is crucial for DCM to determine the required Map Data version of the subsystems. DCM performs a simple version check between existing subsystems Map Data version and the active Map Data version. Depending on the outcome of this version check DCM sends the required Map Data to the subsystems.

Notes:

1. Even though this is an initial request scenario, it involves a possibility to provide an acknowledgement to the subsystems indicating they already have the current Map Data version. This is to ensure that we have a generic version check process in DCM.
2. It is assumed that the subsystems have an existing communication session.

The example sequence below shows how the Map Data is distributed to the RCA subsystems.

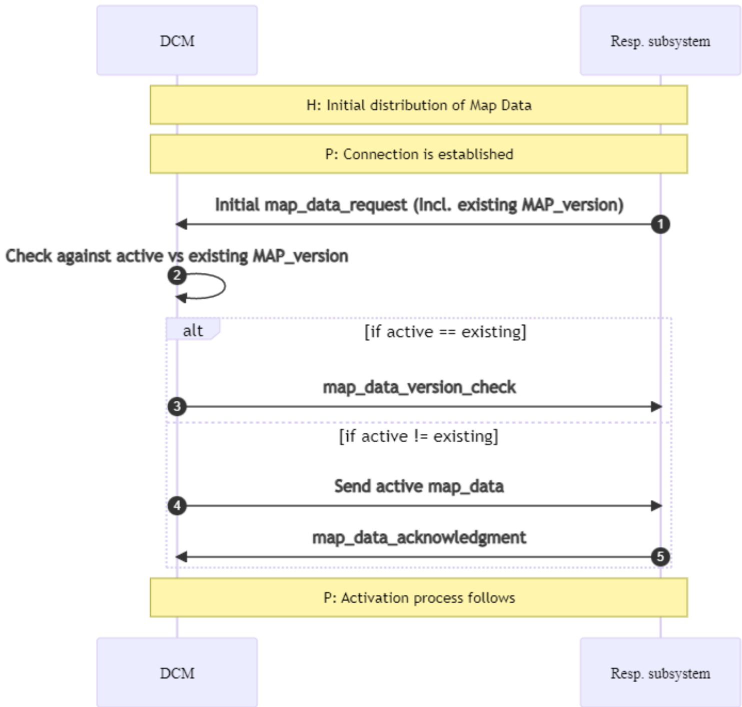

Figure 4: Initial distribution of Map Data

For scenario descriptions refer to Appendix 7.1.1

The sequence diagram uses prefixes such as ‘H’ and ‘P’ to indicate headings and processes respectively.

3.1.1.3. Scenario: Distribution of Map Data update to all trackside subsystems

During distribution of Map Data updates to subsystems, the Map Data is simply pushed to all the subsystems. Unlike the initial distribution case, there is no request from subsystems. This is because DCM (together with TOPO4) is the only subsystem which would be aware of a Map Data update. Hence, the update process is triggered as push aspect from DCM.

Note: It is assumed that the subsystems have an existing communication session.

An example sequence below shows how the Map Data update is distributed to the RCA subsystems.

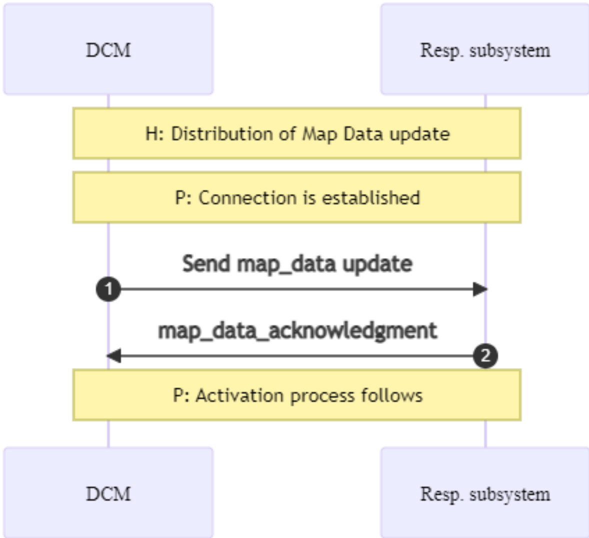

Figure 5: Distribution of Map Data update

For scenario descriptions refer to Appendix 7.1.2

The sequence diagram uses prefixes such as ‘H’ and ‘P’ to indicate headings and processes respectively.

3.1.1.4. Scenario: Restart / Shutdown of a trackside subsystem

During an unintended or intended restart / shutdown of a trackside subsystem, it is imperative that the Map Data within the subsystems are deactivated. This is to ensure that at no point the systems run on outdated Map Data. Consequently, the subsystems request for Map Data along with information on existing Map Data version. This request is similar to the initial Map Data request as mentioned in section 3.1.1.2. Therefore, the principles defined in section 3.1.1.2 also apply here.

Note: It is assumed that the subsystems have an existing communication session.

The example sequence below shows how the actuality of Map Data is checked in the RCA subsystems.

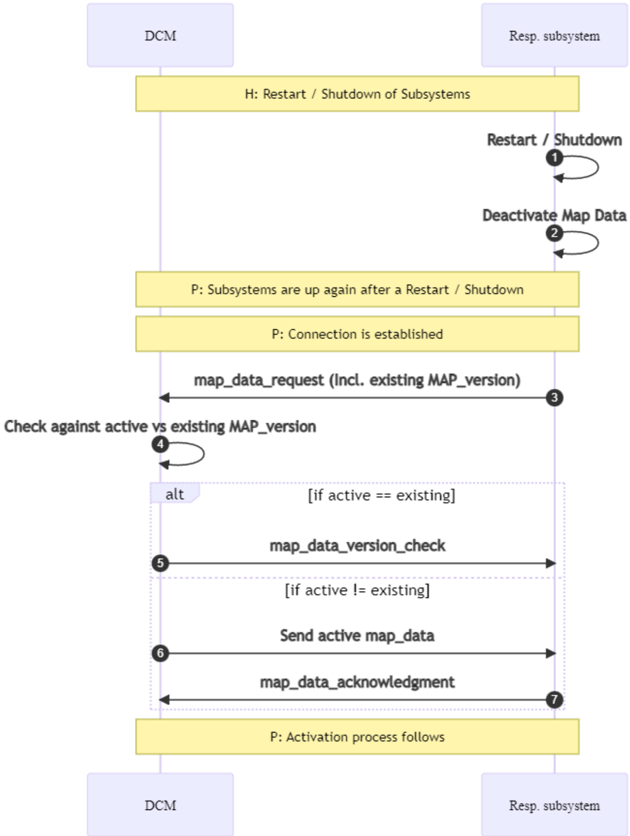

Figure 6: Restart / shutdown of a trackside subsystem

For scenario descriptions refer to Appendix 7.1.3

The sequence diagram uses prefixes such as ‘H’ and ‘P’ to indicate headings and processes respectively.

3.1.2. Distribution of Railway Specific Parameter Data

This section will be further elaborated in future releases.

3.1.3. Distribution of System Configuration Data

For the distribution of Software Configuration Data, a maintenance mode as proposed by EULYNX can be used. Refer to chapter 4.5 for details). Another option that has to be investigated yet is the feasibility of using the same principles as for Map Data.

This section will be further elaborated in future releases.

3.2. Activation principles for Configuration Data

Similar to distribution principles, the activation principles are also divided into three sections to describe the distribution principles for Map Data, Railway Specific Parameter Data, and System Configuration Data.

3.2.1. Activation of Map Data

The activation approach uses the SMI interfaces for the transmission of the activation command for the distributed configurations.

Note: The activation process considers only the “good weather case” (i.e. no failures, perturbations and incidents), meaning that “bad weather cases” are not considered at this point.

Note: The explanations following below refer to APS as one logical component and do not refer to the subsystem's architecture like OA, FOT, MT, MOT.

The activation of Map Data will most likely be initiated by two major actors, namely Infrastructure Manager and Plan Execution/Planning System (PE/PAS).

On the one hand, the Infrastructure Manager (IM) is responsible for activation of the Map Data. The IM authorises the activation of Map Data to DCM by ensuring that the changes to the actual railway infrastructure have been built and approved as per plan and are ready to be operated, also regarding potential other involved non-RCA systems.

On the other hand, PAS via PE is responsible to provide the optimal activation time to DCM. Based on this, PE selects an activation window and ensures that the required safety checks with regards to performing Map Data activation are in place(e.g. elements in the affected area are not claimed by Movement Permissions, further safety checks have to be defined yet).

Based on this activation window, DCM requests APS to prepare the Map Data for activation. APS as an authorising system, is responsible for setting the affected Map Data elements to a restricted state. A restrictive state is any state that blocks elements of Map data to be used in railway operation (i.e. to be used for MPs) during Map Data activation.

Consequently, APS publishes the changes in states via SCI to other subsystems and informs DCM that the area for which the Map Data activation was requested is prepared for activation.

Based on this prepared state of Map Data for activation from APS, DCM provides a synchronised activation command to all the subsystems. This activation command on the one hand triggers APS to check if there are any new elements as part of the input Map Data that are also required to be set to a restricted state. The means how APS will achieve the restricted state e.g., by setting a URA, will be analysed and detailed in a later stage. When present, respective elements are set to a restricted state. These corresponding state changes are also published via SCI to other subsystems.

This activation command on the other hand acts as a trigger within all subsystems to eventually activate the Map Data. Depending on the status of activation, the subsystems provide a Map Data acknowledgement to DCM.

These Map Data acknowledgements are collected by DCM and once the Map Data acknowledgements from all the subsystems are received (considering all acknowledgements are positive), DCM provides a synchronised "activation completed" command to all subsystems.

This "activation completed" command on the one hand ensures to commit the activation process within individual subsystems (bad case scenario: "rollback" is initiated) and on the other hand acts as a trigger for APS to set the affected elements to an operable state. An operable state is any state that allows the elements of Map Data to be used in railway operation post Map Data activation (might involve actions like "clearing the track by first train movement in restricted mode over new/changed track edge").

An example sequence below shows how the activation command is transmitted in a synchronised way through the RCA subsystems.

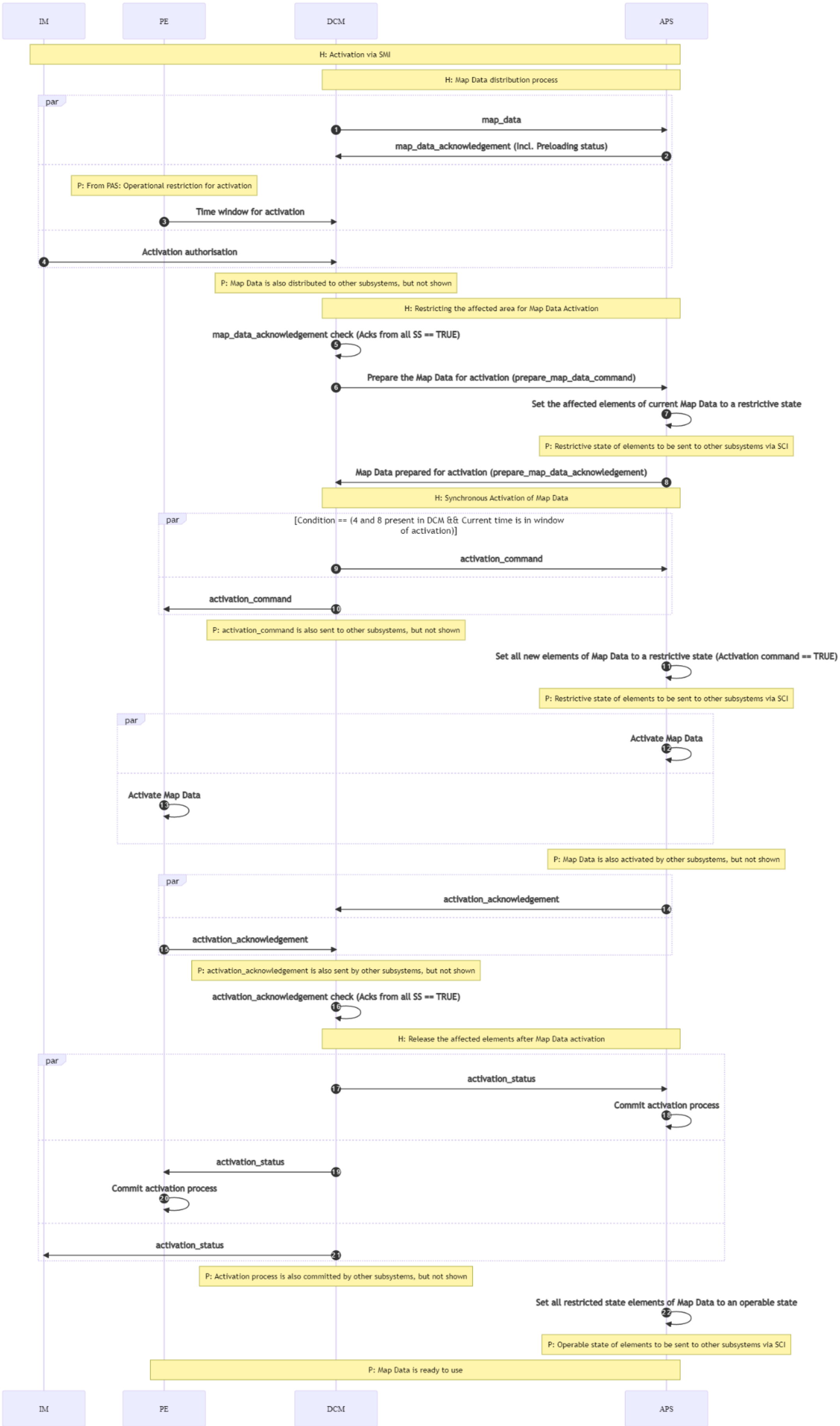

Figure 7: Activation of Map Data via SMI

For scenario description refer to Appendix 7.1.4

Note:

1. For more information on parallel fragments used in above sequence diagram, refer to [11]
2. The sequence diagram uses prefixes such as ‘H’ and ‘P’ to indicate headings and processes respectively.

3.2.2. Activation of Railway Specific Parameter Data

This section will be further elaborated in future releases.

3.2.3. Activation of System Configuration Data

The approach of activating System Configuration Data is to use a maintenance mode. Another option that has to be investigated yet is the feasibility of using the same principles as for Map Data activation.

This section will be further elaborated in future releases.

3.3. Reference implementation

The SMI used in RCA is an evolution of the EULYNX SMI specified in /Eu.Doc.76/ and considers the OC configuration concept. Whereas the EULYNX SMI mainly transfers system data, the RCA SMI also distributes a significant amount of Railway Configuration Data (i.e. Map data and Railway Specific Parameters). Therefore, it is to be expected that there are additional requirements for the DCM in RCA in comparison to the DCM specified by EULYNX resulting in additional requirements and use cases of the RCA SMI.

The EULYNX SMI specification /Eu.Doc.76/ currently includes the service functions:

- Loading procedure for configuration data and engineering data
- Updating specific software of connected systems (not specified yet, only as a placeholder)

The information model and transferred data items specified in the loading procedure are considered in chapter 4.3.2.

The following figure provides the logical (and partially physical) architecture of the OC-Platform (green) and the interface to RCA subsystems (blue).

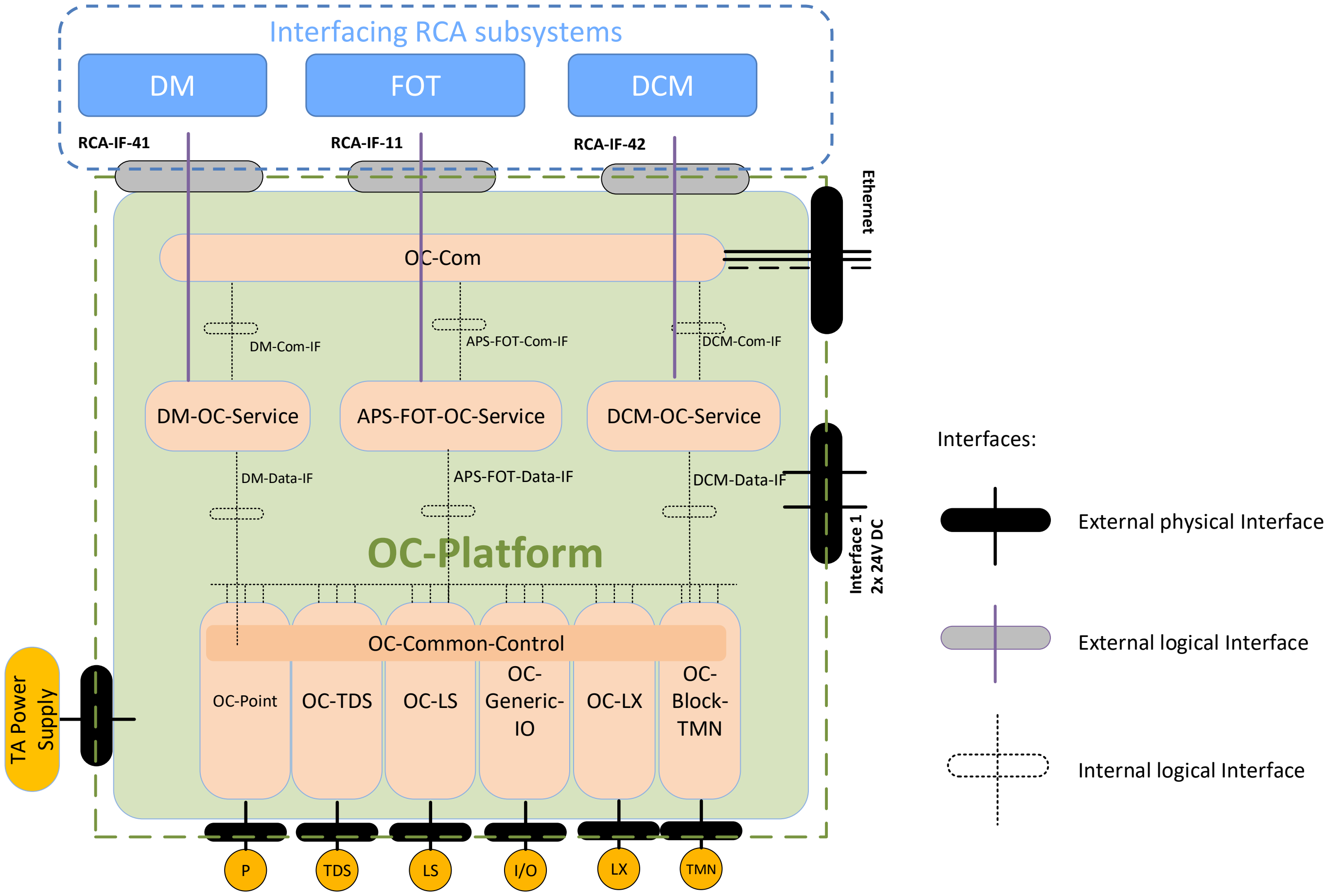

Figure 8: Detailed interface between OC-Platform and RCA architecture

The mapping of the architectures at the interfaces from an EULYNX to an RCA perspective are given in Table 2 below. The additional SMI interfaces that need to be considered in RCA are listed in Table 1 in chapter 2.1.

EULYNX architecture	RCA architecture
Interface RCA-IF-11	(External) interfaces SCI-P, SCI-TDS, SCI-LS, SCI-LX, SCI-IO
Interface RCA-IF-42	SMI
Interface RCA-IF-41	SDI

Table 2: Mapping EULYNX - RCA

4. Structure of SMI

This chapter elaborates the structure of SMI by defining detailed application layer messages. This chapter identifies, classifies, and describes each message that will be transmitted via SMI.

Based on the categorisation of data from chapter 3, the required messages are identified and listed in the table below. These list of messages as a part of application level for SMI.

Message ID	Message name	Source	Sink
1	Map Data	DCM	All RCA trackside sub-systems
2	Map Data acknowledgement	All RCA trackside sub-systems	DCM
3	Map Data preparation command	DCM	APS
4	Map Data preparation acknowledgement	APS	DCM
5	Map Data request	All RCA trackside sub-systems	DCM
6	Map Data version check	DCM	All RCA trackside sub-systems
7	Activation command	DCM	All RCA trackside sub-systems
8	Activation acknowledgement	All RCA trackside sub-systems	DCM
9	Activation status	DCM	All RCA trackside sub-systems
10	Railway specific parameter data	DCM	All RCA trackside sub-systems
11	Railway specific parameter data acknowledgement	All RCA trackside sub-systems	DCM
12	Software data	DCM	All RCA trackside sub-systems
13	Software data acknowledgement	All RCA trackside sub-systems	DCM
14	Security data	DCM	All RCA trackside sub-systems
15	Security data acknowledgement	All RCA trackside sub-systems	DCM
16	Communication parameter data	DCM	All RCA trackside sub-systems
17	Communication parameter data acknowledgement	All RCA trackside sub-systems	DCM

Table 3: List of Messages

The detailed content of upstream and downstream messages of SMI will be covered in the next sections. The next sections of this chapter are divided into the three categories: Railway Configuration Data, activation data for Map, and System Configuration Data. Each section consists of message structure definition for the respective categories.

Note: Not all messages from the table are defined for the current iteration. The detailed content definition is restricted with availability of information on the topic.

4.1. General structure of messages

The content of a message as a part of SMI is conceptually defined as following:

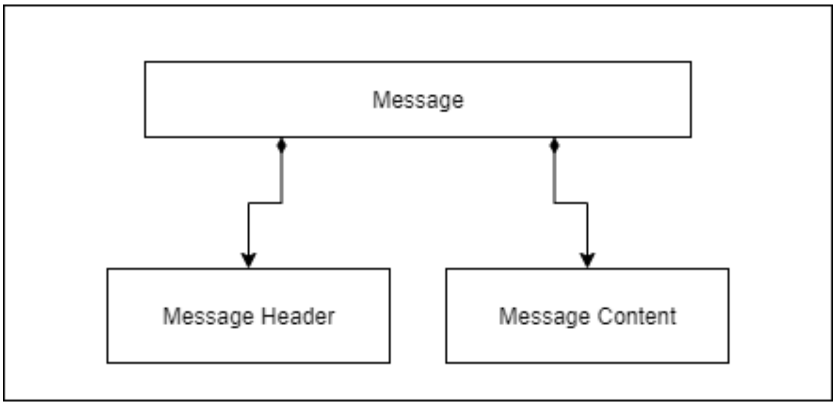

Figure 9: Structure of a Message

Every message consists of a header with predefined information used to identify the message. The header definition remains constant in terms of included parameters. The content of this header is defined in section 4.2

The actual content of a message can vary depending on which message is being transmitted through the interface. The definition of different messages is defined in the next sections.

4.2. Message Header

This section defines the structure of a message header.

Name	Type	Multiplicity	Description
Packet_ID	INT	1	Packet number, used in the header for each packet, allowing the receiving subsystems to identify the data which follows.
length	INT	1	See [5], Section 7.5.1.49
dateTime	INT	1	Date and timestamp of the message.
checksum	Hexa-decimal	1	Value indicating the checksum for the packet.

Table 4: Message header definition

4.3. Distribution of Railway Configuration Data

This section defines the structures of different messages under the scope of railway configuration data.

4.3.1. Map Data

This section defines the structures of different messages under the scope of Map Data.

For definition on Map Data, refer to [2].

4.3.1.4. Domain Object Model of Map Data

The structure of Map Data (on an abstract level) is a collection of instances of relevant objects encompassed by the tiers 1, 2, 3, and 4 of the Map Object Catalogue. The domain object model of Map Data only represents the structural construct of the Map Data along with abstract references to tiers objects. For detailed content and definition of the objects within the tiers refer to MAP Object Catalogue. [3].

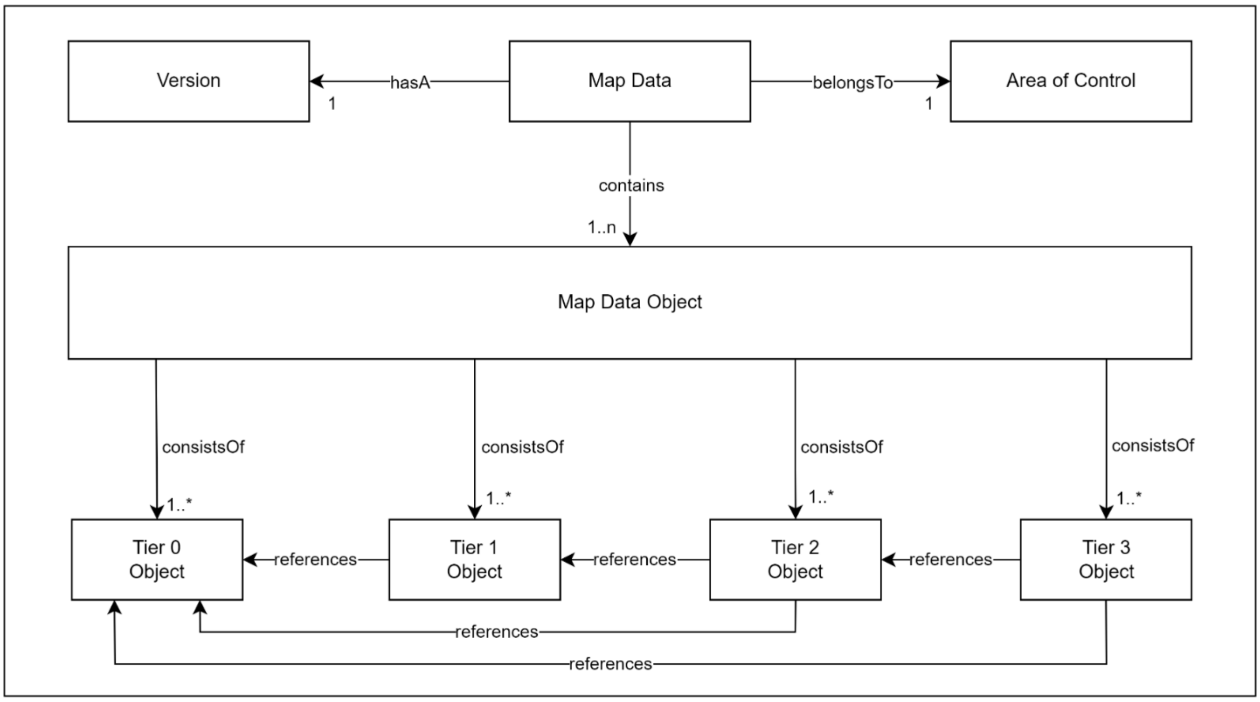

Figure 10: Abstract concept for Map Data

4.3.1.5. Messages

The messages are specified in a tabular format with the columns name, type, multiplicity, and description providing detailed information on the attributes of the message.

4.3.1.5.1. Message: map_data

Description:

map_data represents the interface exchange item which specifies the structure and content of Map Data sent from DCM to all subsystems.

Note: map_data is currently defined as generic content and subsystem-specific message content will be defined in later versions of the document.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, e.g.: Packet id, length, date, time, etc.

			Defined as per Section 4.2
id	HEX(UUID)	1	Unique identifier for Map Data
version	INT	1	Map Data version Information incl. if an object created/modified/deleted as defined in [3]
belongsTo AreaOfControl	*AreaOfControl*	1	Reference to Area of Control as defined in [3]
consistsOfTier0Objects	*(Tier 0 Objects)*	1..*	List of tier 0 objects as defined in [3]
consistsOfTier1Objects	*(Tier 1 Objects)*	1..*	List of tier 1 objects as defined in [3]
consistsOfTier2Objects	*(Tier 2 Objects)*	1..*	List of tier 2 objects as defined in [3]
consistsOfTier3Objects	*(Tier 3 Objects)*	1..*	List of tier 3 objects as defined in [3]
mapDataRelease	*Release*	0..1	Reference to the release information as defined in [3]

Table 5: Map Data definition

4.3.1.5.2. Message: map_data_acknowledgement

Description:

map_data_acknowledgement represents the interface exchange item which specifies the structure and content of acknowledgement message sent as feedback acknowledging the receival and successful preloading of Map Data.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, and time, etc. Defined as per Section 4.2
ID	HEX(UUID)	1	Map Data identifier for which the acknowledgement is provided.
Version	INT	1	Map Data version Information

PL _Status	INT	1	Unique status indicating the status of Map Data preloading e.g. Preloaded or Unable to Preload

Table 6: Map Data acknowledgement definition

4.3.1.5.3. Message: prepare_map_data_command

Description:

prepare_map_data_command represents the interface exchange item which specifies the structure and content of a Map Data preparation command sent from DCM to APS initiating the process to prepare the Map Data for activation.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, time, etc. Defined as per Section 4.2
ID	HEX(UUID)	1	Unique identifier for Map Data
Version	INT	1	Map Data version Information

Table 7: Prepare Map Data command definition

4.3.1.5.4. Message: prepare_map_data_acknowledgement

Description:

prepare_map_data_acknowledgement represents the interface exchange item which specifies the structure and content of a Map Data preparation acknowledgement sent from APS to DCM acknowledging the Map Data are prepared for the activation process.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, time, etc. Defined as per Section 4.2
ID	HEX(UUID)	1	Map Data identifier for which the acknowledgement is provided.
Version	INT	1	Map Data version Information
PREP_Status	INT	1	Unique status indicating the status for preparing the map data e.g. Prepared for activation or Unable to preparation

Table 8: Prepare Map Data acknowledgement definition

4.3.1.5.5. Message: map_data_request

Description:

map_data_request represents the interface exchange item which specifies the structure and content of a Map Data request sent from subsystems to DCM to get Map Data.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, time, etc. Defined as per Section 4.2
Subsystem_ID	HEX(UUID)	1	Unique identifier of the requesting subsystem
ID	HEX(UUID)	1	Unique identifier for Map Data
Version	INT	1	Map Data version Information e.g. currently loaded/activated Map Data version

Table 9: Map Data request definition

Message: map_data_version_check

Description:

map_data_version_check represents the interface exchange item which specifies the structure and content of a Map Data version check performed within DCM to ensure that the subsystems have the actual version of Map Data.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, time, etc. Defined as per Section 4.2
ID	HEX(UUID)	1	Unique identifier for Map Data
Version	INT	1	Map Data version Information e.g. Required Map Version
VER_check	INT	1	Unique status indicating the actuality of the Map Data e.g. Map Data is Up-to-Date

Table 10: Map Data version check definition

4.3.2. Railway Specific Parameter Data

This section defines the structures of different messages under the scope of Railway Specific Parameters.

4.3.2.4. Domain Object Model of Railway Specific Parameters

The structure of Railway Specific Parameters (on an abstract level) has not been defined yet. Therefore, the domain object model will be provided in the later versions of this document.

This section will be further elaborated in future releases.

4.3.2.5. Messages

For now, the messages are defined mostly by a name and description only. In future, the messages will be specified in a tabular format columns name, type, multiplicity, and description providing detailed information on the attributes of the message.

4.3.2.5.1. Message: railway_specific_parameter_data

Note: railway_specific_parameter_data is expected as specific content sent to all subsystems based on their needs. The subsystems-specific message content will be defined in later versions of this document.

Description:

railway_specific_parameter_data represents the interface exchange item with the structure and content of subsystem specific Railway Specific Parameters sent from DCM to all subsystems.

Attributes:

Name	**Type**	**Multiplicity**	**Description / Resolution**
Header		1	Contains required information to identify and interpret transmitted packet, e.g.: Packet id, length, date, time, etc. Defined as per Section 4.2
id	HEX(UUID)	1	Unique identifier for Railway Specific Parameter Data
version	INT	1	Railway Specific Parameter Data version Information
RSP_Status	INT	1	Unique status indicating the nature of Railway Specific Parameter Data
…	…	…	TBD: Railway Specific Parameters to be set in the subsystem

Table 11: Railway Specific Parameter definition

Note: The table above is non-exhaustive and represents an extract of attributes of railway_specific_parameter_data only. It might be updated in a later version of this document.

4.3.2.5.2. Message: railway_specific_parameter_data_acknowledgement

railway_specific_parameter_data_acknowledgement represents the interface exchange item which specifies the structure and content of acknowledgement message sent as feedback acknowledging the receival of Railway Specific Parameters.

Attributes:

Name	Type	Multiplicity	Description / Resolution

Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, time, etc. Defined as per Section 4.2
id	HEX(UUID)	1	Railway Specific Parameter Data identifier for which the acknowledgement is provided.
version	INT	1	Railway Specific Parameter Data version Information
RSP_Status	INT	1	Unique status indicating the status of Railway Specific Parameter Data reception.

Table 12: Railway Specific Parameter data acknowledgement definition

4.4. Activation messages for Map Data

Activation data for Map describes the necessary data required by the subsystems to safely activate the Map Data. This section defines the structures of different messages under the scope of Activation of Map Data

4.4.1. Messages

The messages are specified in a tabular format columns name, type, multiplicity, and description providing detailed information on the attributes of the message.

4.4.1.4. Message: activation_command

Description:

activation_command represents the interface exchange item which specifies the structure and content of an activation command sent from DCM to subsystems to initiate the activation of Map Data within subsystems.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, time, etc. Defined as per Section 4.2
ID	HEX(UUID)	1	Unique identifier for Map Data
Version	INT	1	Map Data version Information

Table 13: Activation command definition

4.4.1.5. Message: activation_acknowledgement

Description:

activation_acknowledgement represents the interface exchange item which specifies the structure and content of an activation acknowledgement sent from subsystems to DCM acknowledging the activation of Map Data.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, time, etc. Defined as per Section 4.2
ID	HEX(UUID)	1	Map Data identifier for which the acknowledgement is provided.
Version	INT	1	Map Data version Information
ACK_Status	INT	1	Unique status indicating the status of activation of Map Data e.g. Activated or Unable to activate

Table 14: Activation acknowledgement definition

Message: activation_status

Description:

activation_status represents the interface exchange item which specifies the structure and content of an activation status sent from DCM to subsystems acknowledging the completion/abortion of the activation process.

Attributes:

Name	Type	Multiplicity	Description / Resolution
Header		1	Contains required information to identify and interpret transmitted packet, i.e.: Packet id, length, date, and time, etc. Defined as per Section 4.2
ID	HEX(UUID)	1	Unique identifier for Map Data
Version	INT	1	Map Data version Information
ACT_status	INT	1	Unique status indicating the status of activation process e.g. Activation completed or Activation abort.

Table 15: Activation status definition

4.5. Distribution of System Configuration Data

The distribution of Software Configuration Data requires a central management of the overall baseline of the system (with different SW versions of included subsystems). The process how a baseline over all subsystems is built has to be established and implemented in one system (DCM would be one option).

There is a rough configuration concept available from EULYNX regarding the OC configuration where the interaction and data/message flows between the involved subsystems were considered. In the table below, these considerations are summarised and a short analysis how they affect the RCA SMI are given.

Note: The software configuration data of other subsystems will be analysed in a later edition of this document once a configuration concept is provided.

Process	Function DCM	Conclusion for RCA SMI
Plan, register, and order OC	Store Map Data Store OC platform configuration data Store OC firmware platform	Transfer Map Data, OC platform configuration data, firmware platform data
Initial installation	-	No transfer of data, OCs are already supplied with the initial installation
Configure OC, connect OC to productive network	Distribute OC configuration Activation of OC configuration (TBD if APS functionality instead) Store “configuration activated” status	Transfer OC configuration and OC activation request
Maintenance	Execute system tests Download new configurations State change to operational/maintenance mode	Transfer OC configuration, OC maintenance mode activation, OC operational mode activation
Operation	Transfer of updateable SW parts during operation: - Application SW - HW/SW compatibility matrix - Communication parameter (to APS) - Initial communication parameter (to DCM) - Authentication certificate OC - Public key APS	Transfer OC SW configuration data

Table 16: OC configuration processes affecting the functionality of DCM and SMI

4.5.1. Messages

This section will be further elaborated in future releases.

5. Cross-cutting issues and NFRs

This section will be further elaborated in future releases.

5.1. Versioning of Messages

This section will be further elaborated in future releases.

5.2. Message Quantity Structure

This section will be further elaborated in future releases.

5.3. Integration of Map Activation time aspect in Messages on SCI-OP

The analysis and integration of the requirements for SCI-OP regarding Map Data activation have to be performed in order to properly consider the time dependency.

This section will be further elaborated in future releases.

6. Open Points

Scope of data

- Specification of subsystem specific Map Data definition for SMI
- Evaluation of the usage of SMI to transfer Configuration Data for On-Board

Versioning

- Analysis how the versioning of Software Configuration Data can be done and how it differs from the versioning of Operational Data
- Specifics on how version check is ensured for Railway System Parameter Data and Software Configuration Data

Data Activation

- Scenario specification on what happens during unsuccessful preloading/activation of Map Data
  a. Determining the reasons to reject preloading/activation
- Specifics about the activation process of Map Data and other Configuration Data
  a. Where lies the safety responsibility?
  b. Analysis if and what safety checks must be performed by APS

Safety and Security

- Safety and Security considerations will be added once there is an overarching RAMSS strategy available

Architecture variations due to migration

- Influence of migration strategies on the Activation aspects of SMI architecture. e.g. Transmission of activation time via ATO-AE to DCM due to unavailability of PE in the first migration phase.

7. Appendix

7.1. Scenario description

This section provides the basic scenario descriptions corresponding to the scenarios defined in this document.

7.1.1. Initial distribution of Map Data to all trackside subsystems

Note: H: Initial distribution of Map Data
Note: P: Connection is established

1. Resp. subsystem ->> DCM: Initial map_data_request (Incl. existing Version)
2. DCM ->> DCM: Check against active vs existing Version

alt if active == existing

3. DCM ->> Resp. subsystem: map_data_version_check

else if active != existing

4. DCM ->> Resp. subsystem: Send active map_data
5. Resp. subsystem ->> DCM: map_data_acknowledgment

end
Note: P: Activation process follows

7.1.2. Distribution of Map Data update to all trackside subsystems

Note: H: Distribution of Map Data update
Note: P: Connection is established

1. DCM ->> Resp. subsystem: Send map_data update
2. Resp. subsystem ->> DCM: map_data_acknowledgment

Note: P: Activation process follows

7.1.3. Restart / Shutdown of a trackside subsystem

Note: H: Restart / shutdown of Subsystems

1. Resp. subsystem ->> Resp. subsystem: Restart / shutdown
2. Resp. subsystem ->> Resp. subsystem: Deactivate Map Data

Note: P: Subsystems is up again after a Restart / shutdown
Note: P: Connection is established

3. Resp. subsystem ->> DCM: map_data_request (Incl. existing Version)
4. DCM ->> DCM: Check against active vs existing Version

alt if active == existing

5. DCM ->> Resp. subsystem: map_data_version_check

else if active != existing

6. DCM ->> Resp. subsystem: Send active map_data
7. Resp. subsystem ->> DCM: map_data_acknowledgment

end
Note: P: Activation process follows

7.1.4. Activation of Map Data

Note: H: Activation via SMI
Note: H: Map Data distribution process
par

1. DCM ->> APS: map_data
2. APS ->> DCM: map_data_acknowledgement (Incl. Preloading status)

and
Note: P: From PAS: Operational restriction for activation

3. PE ->> DCM: Time window for activation

and

4. IM ->> DCM: Activation authorisation

end
Note: P: Map Data is also distributed to other subsystems, but not shown
Note: H: Restricting the affected area for Map Data Activation

5. DCM ->> DCM: map_data_acknowledgement check (Acks from all SS == TRUE)
6. DCM ->> APS: Prepare the Map Data for activation (prepare_map_data_command)
7. APS ->> APS: Set the affected elements of current Map Data to a restrictive state

Note: P: Restrictive state of elements to be sent to other subsystems via SCI

8. APS ->> DCM: Map Data prepared for activation (prepare_map_data_acknowledgement)

Note: H: Synchronous Activation of Map Data
par Condition == (4 and 8 present in DCM && Current time is in window of activation)

9. DCM ->> APS: activation_command

and

10. DCM ->> PE: activation_command

end
Note over DCM: P: activation_command is also sent to other subsystems, but not shown

11. APS ->> APS: Set all new elements of Map Data to a restrictive state (Activation command == TRUE)

Note: P: Restrictive state of elements to be sent to other subsystems via SCI
par

12. APS ->> APS: Activate Map Data

and

13. PE ->> PE: Activate Map Data

end
Note: P: Map Data is also activated by other subsystems, but not shown
par

14. APS ->> DCM: activation_acknowledgement

and

15. PE ->> DCM: activation_acknowledgement

and
Note: P: activation_acknowledgement is also sent by other subsystems, but not shown

16. DCM ->> DCM: activation_acknowledgement check (Acks from all SS == TRUE)

Note: H: Release the affected elements after Map Data activation
par

17. DCM ->> APS: activation_status
18. APS ->> APS: Commit activation process

and

19. DCM ->> PE: activation_status
20. PE ->> PE: Commit activation process

and

21. DCM ->> IM : activation_status

end
Note: P: Activation process is also committed by other subsystems, but not shown

22. APS ->> APS: Set all restricted state elements of Map Data to an operable state

Note: P: Operable state of elements to be sent to other subsystems via SCI
Note: P: Map Data is ready to use

7.2. Business targets and objectives

The following table identifies and lists all of the A.P.M objectives (marked with @ symbol) relevant to SMI, that have been defined and assigned to the various business targets (marked with # symbol) in the A.P.M. objectives document [8].

Business target	Objective
#"Locally scalable" investment for mixed traffic densities	A.P.M.@Deployment of the system must be possible in various configurations but all of them need to fulfil basic requirements
	A.P.M.@Encapsulate minimal viable functionalities in building blocks to enable an individual configuration of the building blocks and simple interfaces
#A.P.M. shall introduce the asset management excellence of digital systems	MO13@Frequently provide Map Data to ensure safety and high-cadence asset modification
	MO16@Allow partial Map Data update
#all railway operations for mainline, regional or urban railways.	APS@Support the configuration of national specific operating rules
#APS shall allow the installation of new network or train elements in small steps "nearly on the fly".	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
	MO01@Provide Map Data versions for testing without impact on operation
	MO04@Efficient distribution of Map Data to Trackside Systems
	O-PE@Support efficient and safe update of Map Data Version during runtime
#A.P.M. shall support the replacement of a large set of trackside assets for large areas (e.g. keeping track of the asset test status over months or supporting forward-and-backward switching of lines)	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
	MO01@Provide Map Data versions for testing without impact on operation
	MO04@Efficient distribution of Map Data to Trackside Systems
	O-PE@Support efficient and safe update of Map Data Version during runtime
#Better traffic flow near to construction sites by seamless integration of mobile / temporary CCS assets	MO04@Efficient distribution of Map Data to Trackside Systems
	MO13@Frequently provide Map Data to ensure safety and high-cadence asset modification
	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
	O-PE@Support efficient and safe update of Map Data Version during runtime
#changes can be done for smaller parts of installations	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on operation
	MO16@Allow partial Map Data update

#data preparation synergy	A.P.M.@All building blocks shall utilise an identical MAP data reference, provided by each MAP data version in order to prevent interpretation efforts
	MO11@Define standard interfaces for provisioning of Map Data to RCA trackside and on-board systems
#faster adaptation of infrastructure to operational needs	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on operation
	MO11@Define standard interfaces for provisioning of Map Data to RCA trackside and on-board systems
#every change of track or CCS functions is proven as handled safe by APS in general.	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
#Allow an instant network wide continuous improvement of the trackside CCS subsystem	MO04@Efficient distribution of Map Data to Trackside Systems
#Flexible safety pattern configuration	APS@Enable individual adaptation of the basic configuration without having to adapt the safety case and the functionality of a component for it
#full automation of infrastructure and vehicle operation	APS@Provide a set of safety functions with a basic configuration that enables safe railway operation for it
#generic risk pattern recognition	APS@Enable individual adaptation of the basic configuration without having to adapt the safety case and the functionality of a component
#interface segregation	APS@Encapsulate minimal viable functionalities in building blocks to enable an individual configuration of the building blocks and simple interfaces
#old compatible with new	MO22@Integrate legacy systems into Trackside Map Data publishing
#only needs actual topology and train information	O-PE@Support efficient and safe update of Map Data Version during runtime
	MO04@Efficient distribution of Map Data to Trackside Systems
#reduce planning, engineering, and authorisation effort by more than 50 %	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
	MO01@Provide Map Data versions for testing without impact on operation
#reduce the technical skill demand for IM	O-PE@Support efficient and safe update of Map Data Version during runtime
#reduction of integration effort	MO10@Ensure modularity and modifiability for MAP systems and interfaces
	MO11@Define standard interfaces for provisioning of Map Data to RCA track-side and on-board systems
#Scheduled speed mix (train mix) and form of usage of a track layout can be changed even on runtime	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
#more assistance	O-PE@Support efficient and safe update of Map Data Version during runtime

#New train types and their specific detailed constraints, properties, weaknesses, or strengths concerning the safe production (e.g. braking capacity) can be introduced to the full network in one (central) step and can be validated or measured on runtime	APS@Enable individual adaptation of the basic configuration without having to adapt the safety case and the functionality of a component for it
#modern and proven ICT architecture, maintenance, and deployment strategies	APS@Allow flexible adaption to data volumes and frequencies without the need to change the code basis APS@Allow flexible adaption to data volumes and frequencies without violating the RAMSS specifications
	O-PE@Support independent updateability of HW and SW and Engineering Data
#Obsolescence shall lead most often to "simple exchange" from the standard interface perspective instead of "change of configuration"	APS@Support exchange of hardware and software with the same functionality treated as 1:1 maintenance with less need of system approval
#Shorter maintenance and construction possessions (blocked tracks) with less impact on the bypassing traffic on other tracks	MO01@Provide Map Data versions for testing without impact on operation
	MO04@Efficient distribution of Map Data to Trackside Systems
	MO14@Avoid the need for end-to-end-testing from APS to trackside assets
	MO16@Allow partial Map Data update
#simple addition of single components under production	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
	MO01@Provide Map Data versions for testing without impact on operation
	MO04@Efficient distribution of Map Data to Trackside Systems
	MO14@Avoid the need for end-to-end-testing from APS to trackside assets
#simple splitting point	MO10@Ensure modularity and modifiability for MAP systems and interfaces
	MO11@Define standard interfaces for provisioning of Map Data to RCA trackside and on-board systems
	APS@The building blocks and their interfaces should have as little version dependency as possible
	O-PE@The building blocks and their interfaces should have as little version dependency as possible
#software upgradeability for central and decentral components	MO13@Frequently provide Map Data to ensure safety and high-cadence asset modification
	MO04@Efficient distribution of Map Data to Trackside Systems
#Standard interface for connecting trackside assets based on EULYNX	MO04@Efficient distribution of Map Data to Trackside Systems
	MO11@Define standard interfaces for provisioning of Map Data to RCA trackside and on-board systems
#standard interfaces to reduce integration effort	MO11@Define standard interfaces for provisioning of Map Data to RCA trackside and on-board systems

#The APS shall support TMS with a detailed information how every part of the network can be used (precise topology state)	MO04@Efficient distribution of Map Data to Trackside Systems
	MO11@Define standard interfaces for provisioning of Map Data to RCA trackside and on-board systems
	O-PE@Support efficient and safe update of Map Data Version during runtime
	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on operation
#reduced onsite testing	MO01@Provide Map Data versions for testing without impact on operation
	MO08@Provide reliable Map Data reflecting the actual topology
	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation
#"on the fly" replacements	O-PE@Support efficient and safe update of Map Data Version during runtime
	APS@Support published changes in Map Data (new, change, delete) during runtime without impact on system operation

Table 17: Business targets and objectives for SMI