

TCCS Configuration - System Requirements

Author(s)	Ghielmetti Cirillo (I-NAT-GST-CCS) , Bois Julien (I-NAT-GST-CCS-EXT - Extern)
Abstract	Definition of the System Requirements for the CCS configuration management process describing the System Requirements at the level of the CCS system.
Classification	public
Status	Open
Version	1.0
Revision	357717
Last Change Date	06.09.2024
Copyright	The reproduction, distribution and utilisation of this document as well as the communication of its contents to others outside EUROPE'S RAIL without express authorisation is prohibited. Offenders will be held liable for the payment of damages. All rights reserved by EUROPE'S RAIL in the event of the grant of a patent, utility model or design.

INFO: History table is not displayed, because this document is in status **doc_open**.

RULE: History table is not displayed, in statuses: { draft doc_open doc_inprogress doc_contentApproval doc_contentDecision }

CONTACT: For more information contact Administrator

Content

Contents

1 Introduction	3
1.1 Document Scope and Purpose	3
1.2 Goals of the Activity	3
2 Input Analysis and References.....	4
3 Glossary.....	6
3.1 Terms and definitions.....	6
3.2 Abbreviations	7
4 System Requirements	8
4.1 Functional requirements	8
4.1.1 System Requirements - CCS System	8
4.1.2 System Requirements - System @ BB Supplier	21
4.2 Non-functional requirements	22
5 Status of the work, open points, issues.....	28
6 Tables	29

1 Introduction

1.1 Document Scope and Purpose

The CCS Configuration Management process describes the conceptual mechanisms to produce, distribute, and activate CCS Configuration items. It details all activities and roles / stakeholders involved in the process.

The CCS Configuration Management Process is intended for distribution of static or semi-static data (like software, parametrisation, data related to topology in interlocking, etc.) related to the railway system. However, the CCS Configuration Management Process does not cover dynamic operational data like Moving Authority (for ETCS) or Journey Profiles (for ATO).

The CCS Configuration Management process is separated and not intended for the maintenance (including update) of virtualised environments. There are commercially available more powerful tools for this specific purpose.

The CCS Configuration Management process is related to CCS components (relevant for the following System Pillar Task 2 domains: Traffic CS, Trackside Asset, Train CS, and Transversal CCS) which may be present in multiple units, either close to infrastructure elements (e.g. CCS trackside instance/area and related signalling components like object controller) or CCS components on-board a train. It may not be a prerequisite that CCS components need to stay in operation when activating / installing [e] a configuration data packet as part of the CCS Configuration Management process. The components may either stay in operation or shut down / restart (basically go out of operation). It depends on the design of the CCS component and what kind of configuration data packet (affecting safety functions and/or operation) is being distributed. It may be a prerequisite that CCS components need to have a limited duration of going out of operation (shut down / restart).

As per the envisaged multi-supplier approach, a CCS Deployment may consist of various separately sourced Building Blocks, integrated, configured, tested, and approved (in line with the CENELEC process) to be deployed according to a well-defined CCS Instance Configuration.

The purpose of this document is to provide the results of the system analysis at CCS level for the configuration management process by means of definition of system requirements. The outcome is based on input documents and experts analyses.

This document will define system requirements from the perspective of different roles (i.e. BB Supplier, Integrator, Operator). These system requirements are the basis to define and standardise (to a certain extent) a configuration management solution that eases the configuration management process for CCS systems (trackside as well as on-board).

Before reading this document, it is recommended to read the context and the basic concepts that drive the CCS configuration management process provided here:

- [!\[\]\(69baca079ef3ab6f03d58fd7e9f950f1_img.jpg\) TCCS Configuration - High Level Concept](#)
- [!\[\]\(2da321c3dc978a55192cb9c452297973_img.jpg\) TCCS Configuration - Operational Epics](#)

1.2 Goals of the Activity

The main activity objectives are the provision of services for configuration management of CCS systems. System Requirements definitions should permit to:

- participate to support configuration management of CCS components, trackside and on-board.
- ease the configuration management of the CCS onboard equipment.
- ease the configuration management of the CCS trackside equipment.

2 Input Analysis and References

Document ID	Document Description	Version
EULYNX Eu.Doc.18	Maintenance and data management specification	4.0
EULYNX Eu.Doc.120	Generic requirements for SMI	1.0
EULYNX Eu.Doc.76	Interface definition and specification SMI	2.0
OCORA-TW07-060	Configuration Management - Concept	1.3
SUBSET-137	On-line Key Management FFFIS	1.0.0
TCCS Configuration - Operational Epics	TCCS Configuration - Operational Epics	
TCCS Configuration - High Level Concept	TCCS Configuration - High Level Concept	

Table 1 List of input documents

In terms of "System Pillar - System Engineering Management Plan (SEMP)" the workitems defined in this document are deliverables of the System Analysis, these are "System Requirements" workitems of the System Analysis as shown in the following picture:

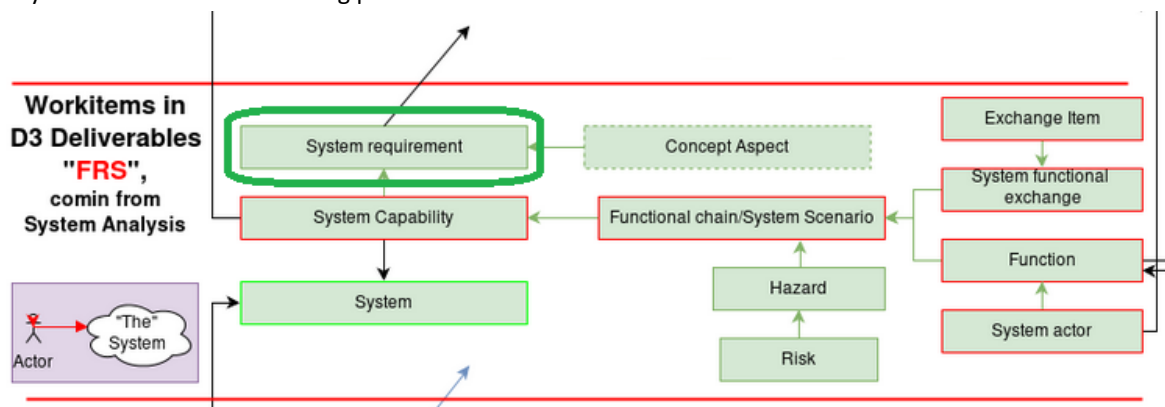


Figure 1 Allocation of the workitems of this document in terms of SEMP

SPT2TS-122388 - Involved systems in the context of the TCCS configuration management process

The overall CCS Configuration Management process can be divided into the following activities: Building Block Realisation, CCS System Instance Integration & Parametrisation, Configuration Distribution, and Configuration Activation within a building block. The overall concept is described in more detail in the referenced document

"TCCS Configuration - High Level Concept". The following diagram shows the different systems involved in the CCS Configuration Management process along with their respective activities executed on these systems. Furthermore, the system boundary of the CCS System from the perspective of the CCS Configuration Management process is illustrated:

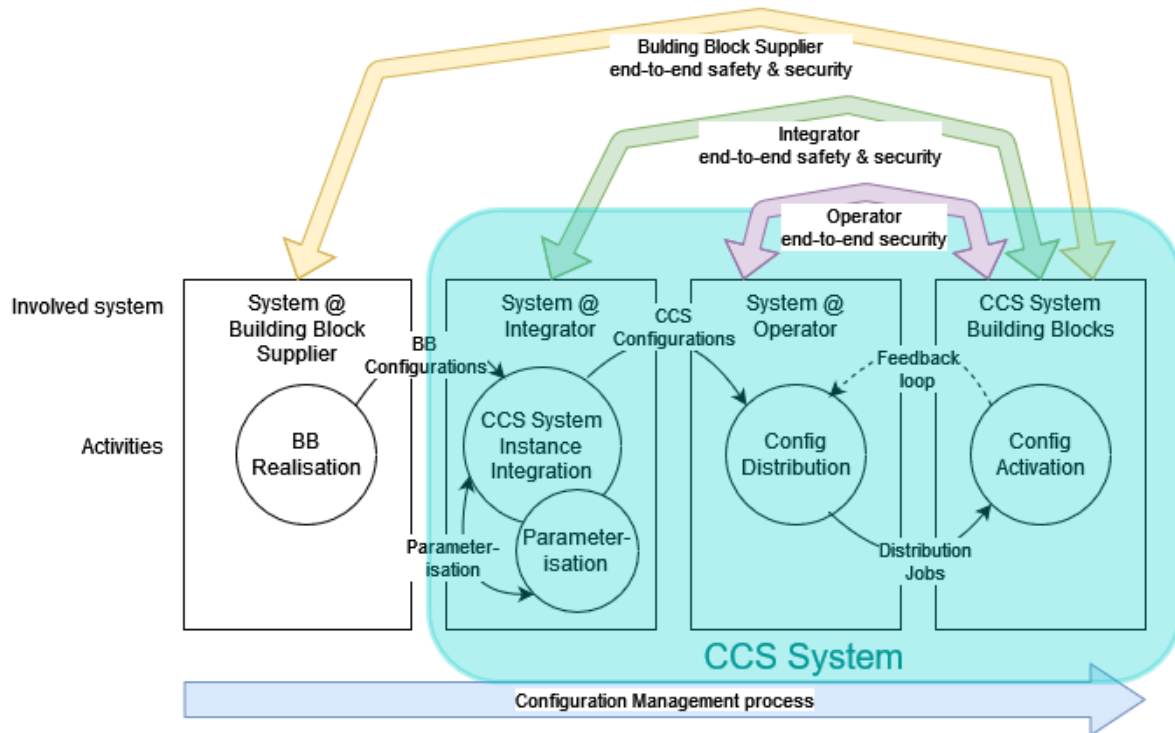


Figure 2 Involved systems in the context of the TCCS configuration management process

The following systems are involved in the configuration management process:

- **System @ Building Block Supplier:** this system is in use at the Building Block Supplier, it allows to prepare, pack and send single new BB Configurations. Each BB Configuration data packet is enriched by the system with specific information to fulfil the needed safety and security measures.
- **System @ Integrator:** this system is in use at the Integrator, it allows to define the needed parametrisation variables of a specific CCS deployment. Each CCS Configuration deployment data packet is enriched by the system with specific information to fulfil the needed safety and security measures.
- **System @ Operator:** this system is in use at the Operator, it allows to schedule the Distribution Jobs. By means of Distribution Jobs single CCS Configuration data packets are distributed to a number of specifically predefined building blocks. Each Distribution Job is enriched by the system with specific information in terms of activation.
- **CCS System building blocks:** these are the single building blocks where a CCS configuration (update data packet) gets activated and installed.
- **CCS System:** the CCS System includes the System @ Integrator, System @ Operator and the CCS System building blocks.

The intention of the SD3 team is to define the CCS Configuration Management (Update) process and the related data distribution mechanisms. It is not intended to fully specify the System @ Integrator or the System @ Operator, only aspects of these systems related to the CCS Configuration Management (Update) process will be addressed.

3 Glossary

3.1 Terms and definitions

Title	Description
Building Block	<p>A Building Block is an equipment based (hardware and/or software) or logical unit of the System having:</p> <ul style="list-style-type: none"> • standardised functionality or aggregates standard functionality it depends on • may have standardised PRAMS requirements (including Tolerable Functional Failure Rate [TFFR]) • may have Safety Integrity Levels [SIL] for functions within the system border and Safety Related Application Conditions [SRAC]) • standardised cyber security requirements (including Security Level [SL] based on the security requirements, and Security Related Application Conditions [SRAC]) • may have (on lower levels) standardised interfaces (on all OSI Layers) towards other Building Blocks and/or external systems. <p>Equipment based Building Blocks are separately sourceable from different suppliers and capable of being integrated by a third party (integrator). A BuildingBlock has one or more BuildingBlockConfigurations. A BuildingBlock must have a unique identifier composed of configurationGroupId and configurationId.</p>
Building Block Configuration	<p>A BuildingBlockConfiguration (BBC) is a node on a layer within the configuration dependency tree.</p> <p>It must be uniquely identifiable within the system and may contain a configurationFile artifact and dependencies to other BBCs.</p> <p>One BuildingBlock (BB) can have one or more BuildingBlock Configurations (BBC).</p> <p>One Building BlockConfiguration (BBC) has exactly one configuration.json file (and a configurationSafe.json if it is a safe BBC).</p> <p>BBCs that itself have no further dependencies in their configuration.json file are the Lowest Updatable Units (LUU - can be updated on its own).</p> <p>BBCs that are updatable must provide a corresponding configurationFile (payload).</p> <p>BBCs that are updatable need an endpoint described in the "configuration.json" file.</p> <p>That BBC endpoint can be accessed using a protocol capable of file transfer (e.g. opc ua).</p>
Building Block Manifest	<p>The Building Block Manifest is a document that describes a Building Block configuration. For more details see the description for SPT2TS-123463 - BuildingBlockConfiguration "configuration.json" document.</p>
CCS Manifest	<p>The CCS Manifest is a document that describes a CCS deployment configuration. For more details see the description for SPT2TS-123463 - BuildingBlockConfiguration "configuration.json" document.</p>
Configuration item	<p>A configuration item refers to any resource within a system that needs to be managed and controlled in order to support the delivery of products or services. It can be an unchangeable item such as a software (provided by the originator it remains unchanged until installation), or it can be a changeable item such as a parametrisation file (provided by the originator it may be changed by a third party before installation).</p> <p>Configuration items are typically identified, documented, and tracked throughout their lifecycle to ensure proper control, maintenance, and change management. They are often</p>

part of a configuration management system or database, which helps in organizing and managing the configuration items and their relationships.
The purpose of managing configuration items is to have a clear understanding of the resources that make up a system, their interdependencies, and their characteristics. This enables effective control, planning, and decision-making, particularly in areas such as asset management, change management, and problem resolution.

Control, Command and Signalling	Control, Command and Signalling
High level BuildingBlocks	<p>Due to the dependency concept the BuildingBlocks can form units of unlimited size. The recursive dependency tree can have any depth. Every level of the dependency tree links to the next level - that allows to assign clear responsibilities for each of the levels.</p> <p>As an example a BuildingBlock "area" might refer to a number of "interlocking interiors" and "filedelements" in its next downstream dependency level.</p> <p>The Top-Level BuildingBlockConfiguration is the root where all dependencies start from.</p>
Operator	An institution operating a system through its life-cycle, in context of the document usually an Infrastructure Manager or Train Operator.
Static (or semi-static) data	<p>Static (or semi-static) data refers to information or data that remains unchanged or constant over time. It is data that does not require frequent updates or modifications. Static data typically includes reference data, constants, configuration settings, or any other data that remains consistent throughout the operation of a system or application. This type of data is often used as a foundation or reference point for various processes or calculations within a system. Static data is generally stored in a read-only format and is not subject to frequent modifications or user interactions. This is often data that requires a homologation process before it can be applied on a system going in operation. However, it can also include data that does not require a homologation like IP-address, security patches, etc.</p> <p>Example of statical data: software, firmware, parametrisation file, data related to the topology stored in an interlocking, braking curves stored in the ETCS on-board, etc.</p>

[9 items found](#) 

3.2 Abbreviations

Abbreviation	Title
BB	Building Block
CCS	Control, Command and Signalling
CI	Configuration item

[3 items found](#) 

4 System Requirements

The requirements listed in this document are defined according to the specifications of the SEMP:

- [SEMP Annex R1 - Requirements patterns syntax](#)
- [SEMP Annex R2 - Rules for writing textual requirements](#)

The requirements defined in this document are valid for the different systems as described in chapter 2.

4.1 Functional requirements

4.1.1 System Requirements - CCS System

SPT2TS-122396 - Provide remote configuration (update) capability

The CCS System shall provide service functions to support the configuration management process from remote (where CCS components are connected to a network and are further away than short walking distance) for CCS building blocks (CCS components being in operation).

Linked Work Items	<p>is derived from: SPT2TS-1392 - Selection of the most suitable distribution channel</p> <p>is derived from: SPT2TS-1393 - Installation of an update without the need of a manual intervention</p> <p>is derived from: SPT2TS-1840 - Operator [III] responsibilities</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>has parent: SPT2TS-122368 - System Requirements - CCS System</p> <p>_ is implemented by: SPT2TS-127206 - Sequence diagram of the configuration update process</p> <p>_ is implemented by: SPT2TS-124931 - Data preparation artifacts</p> <p>_ is implemented by: SPT2TS-126984 - Description of the sequence diagram</p> <p>_ is implemented by: SPT2TS-126747 - Creation of the safe BuildingBlockConfiguration documents and tree structures</p> <p>_ is parent of: SPT2TS-122397 - Provide local configuration (update) capability</p>
-------------------	---

SPT2TS-122397 - Provide local configuration (update) capability

The CCS System shall provide service functions to support the local configuration management process for CCS building blocks, also when no connection is available (no connection between distribution server and CCS building block). For this case the configuration management process might foresee some local activities (staff with its equipment present on-site).

Note: This can be useful for trains in the workshop, where at times not the whole train is powered and therefore no connection from train to off-board might be available to the CCS on-board building blocks.

Linked Work Items	<p>is derived from: SPT2TS-1392 - Selection of the most suitable distribution channel</p> <p>is derived from: SPT2TS-1398 - One single service point on-board</p> <p>is derived from: SPT2TS-1840 - Operator [III] responsibilities</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>has parent: SPT2TS-122396 - Provide remote configuration (update) capability</p> <p>_ is parent of: SPT2TS-122420 - In vehicles only one common local service point for all different on-board systems</p>
-------------------	--

SPT2TS-122420 - In vehicles only one common local service point for all different on-board systems

The CCS System on-board of vehicles shall be connected to the one common local vehicle service point to which all other systems in the vehicle shall also be linked to. This common service point is the access point for local Configuration Management activities.

Note: the one common local service point typically provides gateway/proxy functionality, at least for the CCS on-board building blocks during operation. During first installation/commissioning of the CCS on-board components the situation might be different, as these could not yet properly embedded in their environment.

The one common local service point is used indifferently for safety related and non-safety related functions.

Linked Work Items	<p>is derived from: SPT2TS-1398 - One single service point on-board</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>has parent: SPT2TS-122397 - Provide local configuration (update) capability</p>
-------------------	---

SPT2TS-122390 - Support an easy import of CCS Configurations at Operator

The System @ Operator shall provide an import procedure service function to support and ease the import of CCS Configurations via a standardised interface.

Linked Work Items	<p>is derived from: SPT2TS-1844 - Deployed CCS system instance parametrisation [c]</p> <p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p> <p>is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application</p> <p>is derived from: SPT2TS-49112 - Automatic availability of configuration data packet on maintenance server</p> <p>is derived from: SPT2TS-1395 - Deployment of new keys to RBCs</p> <p>is derived from: SPT2TS-1648 - Deployment of cyber security updates to trackside components in a timely manner</p> <p>is derived from: SPT2TS-1397 - Deployment of new keys to CCS on-board equipment</p> <p>is derived from: SPT2TS-1647 - Deployment of cyber security updates to vehicles in a timely manner</p> <p>is derived from: SPT2TS-127453 - Automatic notification in case of new configuration data packet</p> <p>has parent: SPT2TS-122368 - System Requirements - CCS System</p> <p>_ is parent of: SPT2TS-122418 - Same easy import of CCS Configurations at Operator is applicable to all products</p>
-------------------	---

SPT2TS-122418 - Same easy import of CCS Configurations at Operator is applicable to all products

The System @ Operator shall provide the same import procedure for all different types of CCS building blocks and independently from the Integrator and / or BB Supplier.

Linked Work Items	<p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p> <p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>has parent: SPT2TS-122390 - Support an easy import of CCS Configurations at Operator</p>
-------------------	--

SPT2TS-124162 - Compile Distribution Job

The System @ Operator shall enable the compilation of a Distribution Job based on imported CCS Configurations.

Linked Work Items	<p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>is derived from: SPT2TS-1840 - Operator [III] responsibilities</p> <p>is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application</p> <p>is derived from: SPT2TS-49111 - An update for CCS on-board components is in operational mode within 2 weeks</p> <p>is derived from: SPT2TS-1391 - An update for CCS trackside components is in operational mode within 1 week</p> <p>has parent: SPT2TS-122368 - System Requirements - CCS System</p> <p>_ is implemented by: SPT2TS-125846 - distribution-job.json document</p> <p>_ is derived by: SPT2TS-125967 - Repository contents for one distribution-job</p> <p>_ is implemented by: SPT2TS-125971 - distribution-job configurations</p> <p>_ is implemented by: SPT2TS-125850 - distribution-job.json and distribution-json document semantics</p> <p>_ is implemented by: SPT2TS-125851 - distribution-job.schema.json</p> <p>_ is implemented by: SPT2TS-127125 - distribution.json - the link to BBC configurations</p> <p>_ is parent of: SPT2TS-124161 - Same Distribution Job generation at Integrator is applicable to all products</p> <p>_ is parent of: SPT2TS-124163 - Augment CCS Manifest with Distribution Job parameters</p> <p>_ is parent of: SPT2TS-124164 - Associate Distribution Jobs</p>
-------------------	--

SPT2TS-124163 - Augment CCS Manifest with Distribution Job parameters

The System @ Operator shall provide the functionality to augment the standardised parametrisation variables of the CCS Manifest with values that are specific to each Distribution Job.

Linked Work Items	<p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>is derived from: SPT2TS-1840 - Operator [III] responsibilities</p> <p>is derived from: SPT2TS-1647 - Deployment of cyber security updates to vehicles in a timely manner</p> <p>is derived from: SPT2TS-1648 - Deployment of cyber security updates to trackside components in a timely manner</p> <p>has parent: SPT2TS-124162 - Compile Distribution Job</p> <p>_ is derived by: SPT2TS-125967 - Repository contents for one distribution-job</p> <p>_ is implemented by: SPT2TS-125971 - distribution-job configurations</p>
-------------------	---

SPT2TS-124164 - Associate Distribution Jobs

The System @ Operator shall enable the association of a specific Distribution Job to specific CCS building blocks.

Linked Work Items	<p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>is derived from: SPT2TS-1840 - Operator [III] responsibilities</p> <p>is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet</p> <p>is derived from: SPT2TS-1395 - Deployment of new keys to RBCs</p> <p>is derived from: SPT2TS-1397 - Deployment of new keys to CCS on-board equipment</p> <p>has parent: SPT2TS-124162 - Compile Distribution Job</p>
-------------------	--

SPT2TS-124161 - Same Distribution Job generation at Integrator is applicable to all products

The System @ Operator shall provide the same Distribution Job generation procedure for all different types of CCS building blocks, independently from the BB Supplier or the Integrator.

Linked Work Items	<p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>is derived from: SPT2TS-1840 - Operator [III] responsibilities</p> <p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p> <p>is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application</p> <p>has parent: SPT2TS-124162 - Compile Distribution Job</p> <p>_ is implemented by: SPT2TS-125846 - distribution-job.json document</p>
-------------------	---

SPT2TS-122389 - Support loading procedure for distribution jobs

The System @ Operator shall provide a loading procedure service function to support the provision / deployment of distribution jobs to CCS building blocks via a standardised interface.

Linked Work Items	<p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>is derived from: SPT2TS-1846 - CCS configuration activation [e]</p> <p>is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet</p> <p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p> <p>is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application</p> <p>is derived from: SPT2TS-63817 - No or only loose coupling between configuration data packet release and operational data</p> <p>is derived from: SPT2TS-1391 - An update for CCS trackside components is in operational mode within 1 week</p> <p>is derived from: SPT2TS-1395 - Deployment of new keys to RBCs</p> <p>is derived from: SPT2TS-1648 - Deployment of cyber security updates to trackside components in a timely manner</p> <p>is derived from: SPT2TS-49111 - An update for CCS on-board components is in operational mode within 2 weeks</p> <p>is derived from: SPT2TS-1397 - Deployment of new keys to CCS on-board equipment</p> <p>is derived from: SPT2TS-1647 - Deployment of cyber security updates to vehicles in a timely manner</p> <p>has parent: SPT2TS-122368 - System Requirements - CCS System</p> <p>_ is implemented by: SPT2TS-127204 - The Interface "DI 6"</p> <p>_ is implemented by: SPT2TS-125987 - Distribution Process</p> <p>_ is implemented by: SPT2TS-127200 - Example of a distribution-job and distribution for a trackside area of control</p> <p>_ is implemented by: SPT2TS-127199 - Example of a distribution-job and distribution for a number of vehicles sharing the identical homologation</p> <p>_ is parent of: SPT2TS-122391 - Support generation of many distribution jobs in parallel</p> <p>_ is parent of: SPT2TS-122419 - Same loading procedure is applicable to all products</p>
-------------------	--

SPT2TS-122391 - Support generation of many distribution jobs in parallel

The System @ Operator shall provide a loading procedure service function enabling the provision / deployment of distribution jobs for a number of CCS building blocks in parallel.

Linked Work Items	<p>is derived from: SPT2TS-1389 - Deployment of an update to multiple same CCS components in parallel</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p>
-------------------	--

	<p>is derived from: SPT2TS-2263 - Homogenous configuration versions in the field</p> <p>is derived from: SPT2TS-2254 - Distribution start of a configuration packet in good time</p> <p>is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application</p> <p>is derived from: SPT2TS-1648 - Deployment of cyber security updates to trackside components in a timely manner</p> <p>is derived from: SPT2TS-1647 - Deployment of cyber security updates to vehicles in a timely manner</p> <p>has parent: SPT2TS-122389 - Support loading procedure for distribution jobs</p> <p>_ is implemented by: SPT2TS-125969 - For one homologated model only</p> <p>_ is implemented by: SPT2TS-127199 - Example of a distribution-job and distribution for a number of vehicles sharing the identical homologation</p>
--	---

SPT2TS-122419 - Same loading procedure is applicable to all products

The System @ Operator shall provide the same loading procedure for all different types of CCS building blocks, and independently from the Integrator and / or BB Supplier.

Linked Work Items	<p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p> <p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>has parent: SPT2TS-122389 - Support loading procedure for distribution jobs</p>
-------------------	---

SPT2TS-124027 - Perform automated configuration activation and installation

The building blocks of the CCS System shall automatically start a configuration activation and installation process when the activation point in time is reached, and other defined preconditions for activation are fulfilled.

Note: the building block of the CCS System may synchronise the configuration activation and installation process with its environment, depending on the content of the new configuration (what needs to be installed). This has to be considered in the development of the Configuration Management Process.

Linked Work Items	<p>is derived from: SPT2TS-1846 - CCS configuration activation [e]</p> <p>is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet</p> <p>is derived from: SPT2TS-2252 - Flexible installation of an update: activation time suitable for operations</p> <p>is derived from: SPT2TS-2253 - A configuration packet is aligned with its intended environment</p> <p>is derived from: SPT2TS-1393 - Installation of an update without the need of a manual intervention</p> <p>has parent: SPT2TS-122368 - System Requirements - CCS System</p> <p>_ is implemented by: SPT2TS-127206 - Sequence diagram of the configuration update process</p> <p>_ is implemented by: SPT2TS-126719 - Analysis of the safe BuildingBlockConfiguration tree</p> <p>_ is implemented by: SPT2TS-126720 - Persistent Knowledge about the currently operated safe BuildingBlockConfiguration versions</p> <p>_ is implemented by: SPT2TS-126723 - Feedback: safe Top-Level BBC version switch is possible</p> <p>_ is implemented by: SPT2TS-126721 - Calculating "Difference" in case of a High Level</p>
-------------------	--

	BuildingBlockConfiguration version switch _ is implemented by: SPT2TS-126725 - Provide operational token after successful activation _ is implemented by: SPT2TS-126929 - Update of safety-related data / software _ is implemented by: SPT2TS-126928 - Update of non-safety-related data / software _ is implemented by: SPT2TS-126931 - Update processing _ is implemented by: SPT2TS-126984 - Description of the sequence diagram _ is implemented by: SPT2TS-125970 - distribution-job activation _ is parent of: SPT2TS-124026 - Perform automated configuration activation and installation for a replacement unit _ is parent of: SPT2TS-123910 - Implement standardised high-level modes and states for activation and installation
--	---

SPT2TS-123910 - Implement standardised high-level modes and states for activation and installation

The building blocks of the CCS System shall all implement standardised high-level modes and states to ensure that all building blocks have the same activation and installation behaviour.

Note: the internal technical activation and installation procedure for a CCS configuration (how the technical installation is started and performed within the building block design) is a proprietary implementation that complies with the required safety and security level.

Linked Work Items	is derived from: SPT2TS-1846 - CCS configuration activation [e] is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components is derived from: SPT2TS-2246 - CCS field components only enter into regular operation if these are in a defined proper state is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application has parent: SPT2TS-124027 - Perform automated configuration activation and installation
-------------------	--

SPT2TS-124026 - Perform automated configuration activation and installation for a replacement unit

The building blocks of the CCS System shall automatically start a configuration verification, and if needed a configuration activation and installation process, when a replacement unit (a unit put in operation to replace another defect unit) is installed.

Note-1: a replacement unit from stock might need some basic parametrisation (e.g. setting of the component unique identifier, network settings, etc.) in order to start the verification, configuration activation and installation process.

Note-2: the building block of the CCS System has to synchronise the configuration activation and installation process with its environment.

Linked Work Items	is derived from: SPT2TS-1846 - CCS configuration activation [e] is derived from: SPT2TS-124023 - Automated configuration process for a replacement unit has parent: SPT2TS-124027 - Perform automated configuration activation and installation _ is parent of: SPT2TS-124277 - Perform automated import of basic parametrisation for a replacement unit
-------------------	---

SPT2TS-124277 - Perform automated import of basic parametrisation for a replacement unit

The building blocks of the CCS System being a replacement unit shall implement an automated import function of basic parametrisation values that are valid for the specific environment.

Note: example of basic parametrisation values can be network settings, so that the replacement unit building block can communicate with its new environment.

Linked Work Items	<p>is derived from: SPT2TS-1846 - CCS configuration activation [e]</p> <p>is derived from: SPT2TS-124023 - Automated configuration process for a replacement unit</p> <p>has parent: SPT2TS-124026 - Perform automated configuration activation and installation for a replacement unit</p>
-------------------	---

SPT2TS-124021 - Implement means to ensure consistency within one CCS System instance

The building blocks of the CCS System shall implement standardised means to ensure consistency between the different components of one CCS System instance.

Linked Work Items	<p>is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet</p> <p>is derived from: SPT2TS-2253 - A configuration packet is aligned with its intended environment</p> <p>is derived from: SPT2TS-1393 - Installation of an update without the need of a manual intervention</p> <p>is derived from: SPT2TS-2246 - CCS field components only enter into regular operation if these are in a defined proper state</p> <p>is derived from: SPT2TS-1838 - Integrators [II] responsibilities</p> <p>is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b]</p> <p>has parent: SPT2TS-122368 - System Requirements - CCS System</p> <p>_ is implemented by: SPT2TS-124339 - Handling of dependencies during configuration update</p> <p>_ is implemented by: SPT2TS-126719 - Analysis of the safe BuildingBlockConfiguration tree</p> <p>_ is implemented by: SPT2TS-126720 - Persistent Knowledge about the currently operated safe BuildingBlockConfiguration versions</p> <p>_ is implemented by: SPT2TS-125355 - Releasing High level BuildingBlockConfigurations</p> <p>_ is implemented by: SPT2TS-125359 - Commit Function of the Safe Configuration Authority (SCA)</p> <p>_ is implemented by: SPT2TS-126722 - Confirm incompatible BuildingBlockConfigurations are out of operation</p> <p>_ is implemented by: SPT2TS-126721 - Calculating "Difference" in case of a High Level BuildingBlockConfiguration version switch</p> <p>_ is implemented by: SPT2TS-126728 - Operational View</p> <p>_ is implemented by: SPT2TS-126725 - Provide operational token after successful activation</p> <p>_ is implemented by: SPT2TS-125365 - High Level BuildingBlockConfigurations</p> <p>_ is implemented by: SPT2TS-125364 - Responsibilities on different levels of the dependency tree</p> <p>_ is implemented by: SPT2TS-126930 - System central component</p> <p>_ is implemented by: SPT2TS-126746 - Lease Policy</p> <p>_ is implemented by: SPT2TS-125975 - Safety Attestation</p> <p>_ is implemented by: SPT2TS-126747 - Creation of the safe BuildingBlockConfiguration documents and tree structures</p>
-------------------	--

	<ul style="list-style-type: none"> _ is implemented by: SPT2TS-127245 - Safety attestation during activation and confirmation _ is implemented by: SPT2TS-126687 - Safe BuildingBlockConfigurations: the "configurationSafe.json" documents _ is implemented by: SPT2TS-126686 - Concept of Dependencies _ is derived by: SPT2TS-125357 - BuildingBlock only operational with intended BuildingBlockConfiguration(s) _ is parent of: SPT2TS-124022 - Perform CCS System instance consistency check at runtime
--	--

SPT2TS-124022 - Perform CCS System instance consistency check at runtime

The building blocks of the CCS System shall perform the CCS System instance consistency check also during runtime, when components are in normal operation.

Note: it can occur that updates are installed on the fly (no need for component restart), hence no need to reboot or re-establish any connection.

Linked Work Items	<ul style="list-style-type: none"> is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet is derived from: SPT2TS-2253 - A configuration packet is aligned with its intended environment is derived from: SPT2TS-1393 - Installation of an update without the need of a manual intervention is derived from: SPT2TS-2246 - CCS field components only enter into regular operation if these are in a defined proper state is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b] is derived from: SPT2TS-1838 - Integrators [II] responsibilities has parent: SPT2TS-124021 - Implement means to ensure consistency within one CCS System instance _ is implemented by: SPT2TS-126720 - Persistent Knowledge about the currently operated safe BuildingBlockConfiguration versions _ is implemented by: SPT2TS-126722 - Confirm incompatible BuildingBlockConfigurations are out of operation _ is implemented by: SPT2TS-126721 - Calculating "Difference" in case of a High Level BuildingBlockConfiguration version switch _ is implemented by: SPT2TS-126726 - (Re-)Confirm operational token
-------------------	--

SPT2TS-124205 - Perform a rollback in case of failures during installation

The building blocks of the CCS System shall perform a rollback in case the installation of a new CCS Configuration fails. A rollback means that the CCS building blocks revert back to the original configuration that was present before activation.

Note: a failure during installation could be that the CCS building blocks after having installed the new CCS Configuration cannot come into regular operation. But the failure could also include that some consistency checks of the new CCS Configuration performed before starting installation have failed (despite some checks performed during download), as a consequence the CCS building block remains in the original configuration. The different causes for failure depend on the specific implementation in the different products of the different suppliers.

Linked Work Items	<ul style="list-style-type: none"> is derived from: SPT2TS-1846 - CCS configuration activation [e] is derived from: SPT2TS-2245 - In case of failed installation during configuration process
-------------------	---

	have means to remain in operation has parent: SPT2TS-122368 - System Requirements - CCS System _ is implemented by: SPT2TS-126692 - Rollback
--	--

SPT2TS-122405 - Provide status information for specific distribution jobs

The CCS System shall provide status information for specific distribution jobs, selectable by the user. The following states to be indicated:

- Distribution preparation completed: the server application has completed all activities to prepare the specific CCS Configuration and to send notifications to all involved CCS building blocks about the new CCS Configuration.
- Distribution preparation failed: the server application failed to prepare the specific CCS Configuration or to send the notifications to all involved CCS building blocks.
- CCS Configuration received: each involved CCS building block individually confirms that it has downloaded the specific CCS Configuration which is now available in its internal memory (preloading completed in EULYNX vocabulary).
- CCS Configuration refused: the specific CCS building block refuses to download the specific CCS Configuration.
- CCS Configuration activated: each involved CCS building block individually confirms that it has activated the installation procedure for the specific CCS Configuration.
- CCS Configuration installation successful: each involved CCS building block individually confirms that it has successfully installed the specific CCS Configuration and returned into regular operation.
- CCS Configuration installation failed: the specific CCS building block failed to install the specific CCS Configuration.

Linked Work Items	is derived from: SPT2TS-1840 - Operator [III] responsibilities is derived from: SPT2TS-1845 - CCS configuration distribution [d] is derived from: SPT2TS-1846 - CCS configuration activation [e] is derived from: SPT2TS-1390 - Receive feedback about the status of the configuration (update) process is derived from: SPT2TS-2254 - Distribution start of a configuration packet in good time is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application has parent: SPT2TS-122368 - System Requirements - CCS System _ is implemented by: SPT2TS-126723 - Feedback: safe Top-Level BBC version switch is possible _ is parent of: SPT2TS-122406 - Provide reason for failure related to specific distribution job
-------------------	---

SPT2TS-122406 - Provide reason for failure related to specific distribution job

The CCS System shall provide and log the reason for the failure related to a specific distribution job. This reason is helpful in evaluating and recognising the root cause of the failure, in order to take appropriate action.

Linked Work Items	is derived from: SPT2TS-1840 - Operator [III] responsibilities is derived from: SPT2TS-1845 - CCS configuration distribution [d] is derived from: SPT2TS-1846 - CCS configuration activation [e] is derived from: SPT2TS-1390 - Receive feedback about the status of the configuration (update) process is derived from: SPT2TS-9777 - Want to operate only one Maintenance Server Application has parent: SPT2TS-122405 - Provide status information for specific distribution
-------------------	--

	jobs _ is implemented by: SPT2TS-126724 - Feedback: Status of activations
--	--

SPT2TS-124150 - Be backwards compatible for other than the static configuration data

The building blocks of the CCS System shall be implemented with functions that are backwards compatible in terms of other than the static configuration data (e.g. operational data like ETCS moving authority or ATO journey profile, etc.). This shall also be applicable for new configuration items.

Note: the backwards compatibility ensures no or only loose coupling between configuration items (and the lifecycle of CCS building blocks) and the other data. Without this clear separation the lifecycle management would reach an unmanageable complexity.

Linked Work Items	is derived from: SPT2TS-63817 - No or only loose coupling between configuration data packet release and operational data is derived from: SPT2TS-1838 - Integrators [II] responsibilities is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b] has parent: SPT2TS-122368 - System Requirements - CCS System
-------------------	---

SPT2TS-122399 - Support update of digital certificates

The CCS System shall provide service functions to support the update of digital certificates (used for cyber security purposes) as part of the configuration management process, which is a service available from remote or locally.

Linked Work Items	is derived from: SPT2TS-2250 - Secure installation of an update: configuration process facilitated by security measures is derived from: SPT2TS-1395 - Deployment of new keys to RBCs is derived from: SPT2TS-1397 - Deployment of new keys to CCS on-board equipment is derived from: SPT2TS-1842 - Realisation [a] at Building Block level is derived from: SPT2TS-1838 - Integrators [II] responsibilities is derived from: SPT2TS-1836 - Building Block Supplier [I] responsibility has parent: SPT2TS-122368 - System Requirements - CCS System
-------------------	--

SPT2TS-124025 - Prevent access to secure data stored and used in a failing unit

The building blocks of the CCS System shall provide means to prevent unauthorised access to secure data (key, certificates, etc.) that is stored and used in a unit having a failure and being replaced. The provided means shall prevent that a replaced unit needs to be handled in a secured logistic chain.

Note: Alternatively, it is allowed to introduce a function deleting all secured data (sort of decommissioning process) triggered by trained technicians.

Linked Work Items	is derived from: SPT2TS-124024 - Protection of keys / certificates in defect units is derived from: SPT2TS-1836 - Building Block Supplier [I] responsibility is derived from: SPT2TS-1838 - Integrators [II] responsibilities is derived from: SPT2TS-1842 - Realisation [a] at Building Block level is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b] has parent: SPT2TS-122368 - System Requirements - CCS System
-------------------	--

SPT2TS-122401 - Support an easy import of BB Configurations at Integrator

The System @ Integrator shall provide an import procedure service function to support and ease the import of new BB Configurations from the BB Supplier via a standardised interface.

Linked Work Items	<p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>is derived from: SPT2TS-1838 - Integrators [II] responsibilities</p> <p>is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b]</p> <p>is derived from: SPT2TS-49112 - Automatic availability of configuration data packet on maintenance server</p> <p>is derived from: SPT2TS-49113 - Automatic distribution of data packet from building block supplier</p> <p>is derived from: SPT2TS-127453 - Automatic notification in case of new configuration data packet</p> <p>has parent: SPT2TS-122368 - System Requirements - CCS System</p> <p>_ is parent of: SPT2TS-122415 - Same easy import of BB Configurations at Integrator is applicable to all products</p> <p>_ is parent of: SPT2TS-124129 - Verify BB Configuration from BB Supplier</p>
-------------------	---

SPT2TS-122415 - Same easy import of BB Configurations at Integrator is applicable to all products

The System @ Integrator shall provide the same import procedure for all different types of CCS building blocks and independently from the BB Supplier.

Linked Work Items	<p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p> <p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>has parent: SPT2TS-122401 - Support an easy import of BB Configurations at Integrator</p>
-------------------	---

SPT2TS-124129 - Verify BB Configuration from BB Supplier

The System @ Integrator shall implement BB Configurations integrity and authenticity verification. The scope is that the System @ Integrator can verify the integrity and authenticity (trusted source) of BB Configuration before importing and further processing it. This is especially relevant for parametrisation files where the content is modified by the Integrator.

Linked Work Items	<p>is derived from: SPT2TS-49113 - Automatic distribution of data packet from building block supplier</p> <p>is derived from: SPT2TS-2251 - Safe installation of an update: configuration process facilitated by safety measures</p> <p>is derived from: SPT2TS-12680 - Installation of unaltered update: configuration process facilitated by integrity check measures</p> <p>is derived from: SPT2TS-2250 - Secure installation of an update: configuration process facilitated by security measures</p> <p>is derived from: SPT2TS-1838 - Integrators [II] responsibilities</p> <p>is derived from: SPT2TS-1842 - Realisation [a] at Building Block level</p> <p>has parent: SPT2TS-122401 - Support an easy import of BB Configurations at Integrator</p> <p>_ is implemented by: SPT2TS-125986 - Semantics of the attributes of the BuildingBlockConfiguration "configuration.json" document</p> <p>_ is implemented by: SPT2TS-126689 - Identification of BuildingBlockConfiguration on different levels of the dependency tree</p> <p>_ is implemented by: SPT2TS-125713 - Digital Signatures: ensuring the the</p>
-------------------	--

	authenticity of configuration documents _ is implemented by: SPT2TS-125640 - Ensuring the integrity of the "configuration.json" file and its dependencies
--	--

SPT2TS-122402 - Support an easy export of CCS Configurations at Integrator

The System @ Integrator shall provide an export procedure service function to support and ease the distribution of CCS Configurations from the Integrator to the Operator via a standardised interface.

Linked Work Items	is derived from: SPT2TS-1838 - Integrators [II] responsibilities is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b] is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process is derived from: SPT2TS-49112 - Automatic availability of configuration data packet on maintenance server has parent: SPT2TS-122368 - System Requirements - CCS System _ is implemented by: SPT2TS-127127 - configuration.schema.json _ is parent of: SPT2TS-122417 - Same easy export of CCS Configurations at Integrator is applicable to all products
-------------------	---

SPT2TS-122417 - Same easy export of CCS Configurations at Integrator is applicable to all products

The System @ Integrator shall provide the same export procedure for all different types of CCS building blocks and independently from the BB Supplier.

Linked Work Items	is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process has parent: SPT2TS-122402 - Support an easy export of CCS Configurations at Integrator _ is implemented by: SPT2TS-127127 - configuration.schema.json
-------------------	---

SPT2TS-123904 - Compile CCS Configuration

The System @ Integrator shall enable the compilation of a CCS Configuration based on imported BB Configurations.

Linked Work Items	is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b] is derived from: SPT2TS-49112 - Automatic availability of configuration data packet on maintenance server is derived from: SPT2TS-49113 - Automatic distribution of data packet from building block supplier is derived from: SPT2TS-1838 - Integrators [II] responsibilities is derived from: SPT2TS-127453 - Automatic notification in case of new configuration data packet has parent: SPT2TS-122368 - System Requirements - CCS System _ is implemented by: SPT2TS-127244 - Data Preparation for a logical component _ is implemented by: SPT2TS-126686 - Concept of Dependencies _ is parent of: SPT2TS-122404 - Augment CCS Configuration of Integrator with a CCS Manifest _ is parent of: SPT2TS-123907 - Implement CCS Configuration verification and validation _ is parent of: SPT2TS-123905 - Associate CCS Configuration
-------------------	--

SPT2TS-123905 - Associate CCS Configuration

The System @ Integrator shall enable the association of a specific CCS parameter package to a specific CCS Configuration.

Linked Work Items	<p>is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b]</p> <p>is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet</p> <p>is derived from: SPT2TS-2253 - A configuration packet is aligned with its intended environment</p> <p>is derived from: SPT2TS-1838 - Integrators [II] responsibilities</p> <p>is derived from: SPT2TS-1389 - Deployment of an update to multiple same CCS components in parallel</p> <p>has parent: SPT2TS-123904 - Compile CCS Configuration</p> <p>_ is implemented by: SPT2TS-124962 - Building BlockConfiguration "configuration.json" documents</p>
-------------------	--

SPT2TS-123907 - Implement CCS Configuration verification and validation

The System @ Integrator shall implement CCS Configuration verification and validation capabilities. The scope is to ensure that parametrisation is correct (verification that the correct parametrisation values are defined in a configuration item, e.g. a parametrisation file).

Linked Work Items	<p>is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b]</p> <p>is derived from: SPT2TS-123675 - SUS-054</p> <p>is derived from: SPT2TS-123676 - SUS-055</p> <p>is derived from: SPT2TS-123677 - SUS-056</p> <p>is derived from: SPT2TS-123678 - SUS-057</p> <p>is derived from: SPT2TS-2253 - A configuration packet is aligned with its intended environment</p> <p>is derived from: SPT2TS-1838 - Integrators [II] responsibilities</p> <p>is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet</p> <p>has parent: SPT2TS-123904 - Compile CCS Configuration</p> <p>_ is implemented by: SPT2TS-127245 - Safety attestation during activation and confirmation</p>
-------------------	---

SPT2TS-122404 - Augment CCS Configuration of Integrator with a CCS Manifest

The System @ Integrator shall provide the functionality to augment each CCS Configuration with a specific CCS Manifest that is standardised.

Note-1: the specific CCS Manifest shall include metadata used by the CCS building blocks during the activation and installation process allowing the verification of the CCS configuration's integrity and authenticity, also with respect to safety, security and correct recipient.

Note-2: the aim of a CCS Configuration including an overarching CCS Manifest is to approve (in line with the CENELEC process and for the suitable SIL) once a complete and generic CCS instance deployment, then deploy it multiple times, without the need to follow again the whole assessment process for each specific CCS instance deployment (following the concept of a "reference system" in CSM-RA). For instance, this could be applied to a vehicle fleet, where the same CCS Configuration is deployed to all vehicles of a specific fleet while only changing the CCS instance deployment identifier.

Linked Work Items	is derived from: SPT2TS-1838 - Integrators [II] responsibilities
-------------------	--

	<p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>is derived from: SPT2TS-1388 - Correct activation of an update: correct CCS field component, correct configuration data packet</p> <p>is derived from: SPT2TS-12680 - Installation of unaltered update: configuration process facilitated by integrity check measures</p> <p>is derived from: SPT2TS-2253 - A configuration packet is aligned with its intended environment</p> <p>is derived from: SPT2TS-2251 - Safe installation of an update: configuration process facilitated by safety measures</p> <p>has parent: SPT2TS-123904 - Compile CCS Configuration</p> <p>_ is implemented by: SPT2TS-125986 - Semantics of the attributes of the BuildingBlockConfiguration "configuration.json" document</p> <p>_ is implemented by: SPT2TS-127244 - Data Preparation for a logical component</p> <p>_ is implemented by: SPT2TS-125713 - Digital Signatures: ensuring the the authenticity of configuration documents</p> <p>_ is implemented by: SPT2TS-125640 - Ensuring the integrity of the "configuration.json" file and its dependencies</p> <p>_ is implemented by: SPT2TS-125972 - Safe BuildingBlockConfiguration attributes</p> <p>_ is implemented by: SPT2TS-125852 - BuildingBlockConfiguration "configuration.json" syntax</p>
--	--

4.1.2 System Requirements - System @ BB Supplier

SPT2TS-122400 - Support an easy export of BB Configurations at BB Supplier

The System @ BB Supplier shall provide an export procedure service function to support and ease the distribution of BB Configurations from the BB Supplier to the Integrator via a standardised interface.

Linked Work Items	<p>is derived from: SPT2TS-1836 - Building Block Supplier [I] responsibility</p> <p>is derived from: SPT2TS-1842 - Realisation [a] at Building Block level</p> <p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>is derived from: SPT2TS-49112 - Automatic availability of configuration data packet on maintenance server</p> <p>is derived from: SPT2TS-49113 - Automatic distribution of data packet from building block supplier</p> <p>has parent: SPT2TS-122369 - System Requirements - System @ BB Supplier</p> <p>_ is implemented by: SPT2TS-127203 - The Repository implements the DI 1 interface</p> <p>_ is derived by: SPT2TS-125844 - configurationGroupId.configurationId.configurationVersion form a hierarchy</p> <p>_ is implemented by: SPT2TS-124929 - BuildingBlockConfiguration "configuration.json" document</p> <p>_ is parent of: SPT2TS-122416 - Same easy export of BB Configurations at BB Supplier is applicable to all products</p>
-------------------	---

SPT2TS-122416 - Same easy export of BB Configurations at BB Supplier is applicable to all products

The System @ BB Supplier shall provide the same export procedure for all different types of CCS building blocks and independently from the BB Supplier.

Linked Work Items	<p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p>
-------------------	---

	<p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>has parent: SPT2TS-122400 - Support an easy export of BB Configurations at BB Supplier</p> <p>_ is derived by: SPT2TS-127128 - Immutability</p> <p>_ is derived by: SPT2TS-125844 - configurationGroupId.configurationId.configurationVersion form a hierarchy</p> <p>_ is derived by: SPT2TS-125965 - Updates</p> <p>_ is derived by: SPT2TS-125966 - Repository contents for one BuildingBlockConfiguration</p>
--	---

SPT2TS-122403 - Augment BB configuration item of BB Supplier with a BB Manifest

The System @ BB Supplier shall augment each BB Configuration with a specific BB Manifest.

Note: the specific BB Manifest shall include metadata used by the CCS building blocks during the activation and installation process allowing the verification of the BB Configuration's integrity and authenticity, also with respect to safety and security.

Linked Work Items	<p>is derived from: SPT2TS-1829 - Basic activities of the CCS Configuration Management process</p> <p>is derived from: SPT2TS-1836 - Building Block Supplier [I] responsibility</p> <p>is derived from: SPT2TS-2250 - Secure installation of an update: configuration process facilitated by security measures</p> <p>is derived from: SPT2TS-12680 - Installation of unaltered update: configuration process facilitated by integrity check measures</p> <p>is derived from: SPT2TS-2251 - Safe installation of an update: configuration process facilitated by safety measures</p> <p>has parent: SPT2TS-122369 - System Requirements - System @ BB Supplier</p> <p>_ is derived by: SPT2TS-125966 - Repository contents for one BuildingBlockConfiguration</p> <p>_ is implemented by: SPT2TS-127244 - Data Preparation for a logical component</p> <p>_ is implemented by: SPT2TS-126685 - Specific to logical components and/or hardware models</p> <p>_ is implemented by: SPT2TS-125713 - Digital Signatures: ensuring the the authenticity of configuration documents</p>
-------------------	---

4.2 Non-functional requirements

SPT2TS-122398 - Ensure only authorised staff has system access

The CCS System shall ensure that only authorised staff has access to the system.

Note: authorised staff means that the person has been authenticated, and can only perform granted actions. The latter is according to his role or personal settings. Of course, it is assumed that specific actions are only granted once the person has been properly trained for it, also in terms of safety and security.

Linked Work Items	<p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>is derived from: SPT2TS-2250 - Secure installation of an update: configuration process facilitated by security measures</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p> <p>_ is related to: SPPRAMSS-2311 - The Secure Component shall enforce access based on</p>
-------------------	---

	retrieved permissions for all...
--	--

SPT2TS-122393 - Ensure only configuration data packets from trusted sources are activated and installed

The building blocks of the CCS System shall implement means aligned with the BB Supplier to evaluate during the activation and installation process the BB Manifest, in order to ensure the configuration item's integrity and authenticity, also with respect to safety and cyber-security.

This requirement is valid for binary files that are not enhanced by the Integrator. Such configuration items are transparently transmitted from the BB Supplier to the building block where they are finally installed.

Note: this allows to verify that the BB Configuration is originated from a trusted source.

Linked Work Items	<p>is derived from: SPT2TS-1395 - Deployment of new keys to RBCs</p> <p>is derived from: SPT2TS-1648 - Deployment of cyber security updates to trackside components in a timely manner</p> <p>is derived from: SPT2TS-1397 - Deployment of new keys to CCS on-board equipment</p> <p>is derived from: SPT2TS-1647 - Deployment of cyber security updates to vehicles in a timely manner</p> <p>is derived from: SPT2TS-2250 - Secure installation of an update: configuration process facilitated by security measures</p> <p>is derived from: SPT2TS-1836 - Building Block Supplier [I] responsibility</p> <p>is derived from: SPT2TS-1842 - Realisation [a] at Building Block level</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p> <p>_ is implemented by: SPT2TS-127244 - Data Preparation for a logical component</p> <p>_ is implemented by: SPT2TS-125713 - Digital Signatures: ensuring the the authenticity of configuration documents</p> <p>_ is derived by: SPT2TS-126691 - Authenticity and Integrity of configurations</p>
-------------------	---

SPT2TS-122394 - Ensure only authorised configuration data packets are activated and installed

The building blocks of the CCS System shall implement means standardised with the Integrator to evaluate during the activation and installation process the CCS Manifest, in order to ensure the configuration's integrity and authenticity, also with respect to safety and cyber-security.

This requirement is valid for parametrisation files that have been enhanced with valid parametrisation values by the Integrator.

Note: this allows to verify that the CCS Configuration is originated from an authenticated source, and that the addressed CCS building block is the correct recipient.

Linked Work Items	<p>is derived from: SPT2TS-1395 - Deployment of new keys to RBCs</p> <p>is derived from: SPT2TS-1648 - Deployment of cyber security updates to trackside components in a timely manner</p> <p>is derived from: SPT2TS-1397 - Deployment of new keys to CCS on-board equipment</p> <p>is derived from: SPT2TS-1647 - Deployment of cyber security updates to vehicles in a timely manner</p> <p>is derived from: SPT2TS-1838 - Integrators [II] responsibilities</p> <p>is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b]</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p> <p>_ is implemented by: SPT2TS-125713 - Digital Signatures: ensuring the the authenticity of configuration documents</p> <p>_ is implemented by: SPT2TS-125640 - Ensuring the integrity of the "configuration.json" file and its dependencies</p>
-------------------	---

SPT2TS-122395 - Freedom from interference and documented separation of safety related and non-safety related functions

The building blocks of the CCS System shall be designed and documented with a clear separation between safety related and non-safety related functions (including security functions, and others), where non-safety related functions have a freedom from interference to safety related functions.

Note: the clear separation allows to quickly and easily deploy updates for non-safety related functions (e.g. security patches, error corrections, non-safety related functional enhancements, etc.).

Linked Work Items	<p>is derived from: SPT2TS-1648 - Deployment of cyber security updates to trackside components in a timely manner</p> <p>is derived from: SPT2TS-1647 - Deployment of cyber security updates to vehicles in a timely manner</p> <p>is derived from: SPT2TS-2244 - Same configuration process applicable to all CCS field components</p> <p>is derived from: SPT2TS-1844 - Deployed CCS system instance parametrisation [c]</p> <p>is derived from: SPT2TS-1843 - Deployed CCS system instance integration [b]</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p> <p>_ is implemented by: SPT2TS-127124 - Semantics of the attributes of the BuildingBlockConfiguration "configurationSafe.json" document</p> <p>_ is implemented by: SPT2TS-126927 - Common update mechanism</p> <p>_ is implemented by: SPT2TS-126747 - Creation of the safe BuildingBlockConfiguration documents and tree structures</p> <p>_ is implemented by: SPT2TS-127126 - configurationSafe.schema.json</p> <p>_ is implemented by: SPT2TS-126687 - Safe BuildingBlockConfigurations: the "configurationSafe.json" documents</p> <p>_ is implemented by: SPT2TS-127125 - distribution.json - the link to BBC configurations</p> <p>_ is derived by: SPT2TS-126690 - Safe BuildingBlockConfiguration</p>
-------------------	--

SPT2TS-124187 - Distribution of a new CCS Configuration to a CCS trackside building block takes less than 1 week

The distribution of a new CCS Configuration (Distribution Job) to a specific CCS trackside building block shall take less than 1 week time.

Note: maximum duration of 1 week for distribution of a new CCS Configuration after release for mass distribution is based on some suppositions:

- The CCS trackside building blocks are always in operation (24/7), only exceptionally not operational.
- The 1 week is assumed for data distribution from the start of distribution at the centralised server of the Operator (click on the button) until the new CCS Configuration is available, stored on a CCS trackside building block.

Linked Work Items	<p>is derived from: SPT2TS-1391 - An update for CCS trackside components is in operational mode within 1 week</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p> <p>_ is parent of: SPT2TS-124204 - Scalable distribution process for new CCS Configuration to CCS trackside building blocks</p>
-------------------	---

SPT2TS-124204 - Scalable distribution process for new CCS Configuration to CCS trackside building blocks

The distribution process for new CCS Configuration to CCS trackside building blocks and its performance requirement shall be scalable. This means the time requirement for the update to be in operation is the same independently of the number of components for one CCS instance: 1 or 10 or 300 or more CCS trackside building blocks.

Linked Work Items	<p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>is derived from: SPT2TS-1391 - An update for CCS trackside components is in operational mode within 1 week</p> <p>has parent: SPT2TS-124187 - Distribution of a new CCS Configuration to a CCS trackside building block takes less than 1 week</p>
-------------------	--

SPT2TS-124190 - Activation / installation of a new CCS Configuration in a CCS trackside building block takes less than 2 hours.

The trackside building blocks of the CCS system shall activate / install a new CCS Configuration and go into operation in less than 2 hours.

Note I: maximum duration of 2 hours is based on some suppositions:

- The new CCS Configuration has been successfully distributed and is available to the CCS trackside building block.
- The CCS trackside building block is in operation and basically ready to install the new CCS Configuration, no failures or other issues persist.
- The 2 hours are assumed from the activation point in time defined in the Distribution Job parametrisation until the CCS trackside building block goes into operation.
- The environment of the CCS trackside building block is ready to install the new CCS Configuration.

Note II: In case of activation / installation failure (for whatever reason), latest after 2 hours without update from the application the Operator can assume that the activation / installation failed and react accordingly.

Linked Work Items	<p>is derived from: SPT2TS-1391 - An update for CCS trackside components is in operational mode within 1 week</p> <p>is derived from: SPT2TS-1846 - CCS configuration activation [e]</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p>
-------------------	---

SPT2TS-124191 - Distribution of a new CCS Configuration to a CCS on-board building block takes less than 10 days

The distribution of a new CCS Configuration (Distribution Job) to a specific CCS on-board building block shall take less than 10 days time.

Note: maximum duration of 10 days for distribution of a new CCS Configuration after release for mass distribution is based on some suppositions:

- The CCS on-board building blocks are going into operation at least once for a minimum duration of 12h during the 10 days time. The 12h duration for operation to be counted not earlier than 24h after start of distribution at the centralised server (click on the button).
Rationale is that possibly CCS on-board building blocks are not always in operation (not 24/7).
- The 10 days are assumed for data distribution from the start of distribution at the centralised server (click on the button) until the new CCS Configuration is available, stored on a CCS on-board building block.

Linked Work Items	<p>is derived from: SPT2TS-49111 - An update for CCS on-board components is in operational mode within 2 weeks</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p> <p>_ is parent of: SPT2TS-124203 - Scalable distribution process for new CCS Configuration to CCS on-board building blocks</p>
-------------------	---

SPT2TS-124203 - Scalable distribution process for new CCS Configuration to CCS on-board building blocks

The distribution process for new CCS Configuration to CCS on-board building blocks and its performance requirement shall be scalable. This means the time requirement for the update to be in operation is the same independently of the fleet size: 1 or 10 or 300 or more vehicles.

Linked Work Items	<p>is derived from: SPT2TS-49111 - An update for CCS on-board components is in operational mode within 2 weeks</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p> <p>has parent: SPT2TS-124191 - Distribution of a new CCS Configuration to a CCS on-board building block takes less than 10 days</p>
-------------------	---

SPT2TS-124192 - Activation / installation of a new CCS Configuration in a CCS on-board building block takes less than 2 hours.

The on-board building blocks of the CCS system shall activate / install a new CCS Configuration and go into operation in less than 2 hours.

Note I: maximum duration of 2 hours is based on some suppositions:

- The new CCS Configuration has been successfully distributed and is available to the CCS on-board building block.
- The 2 hours are assumed from the activation point in time defined in the Distribution Job parametrisation until the CCS on-board building block goes into operation.
- The CCS on-board building block is in operation and basically ready to install the new CCS Configuration, no failures or other issues persist.
- The environment of the CCS on-board building block is ready to install the new CCS Configuration.

Note II: In case of activation / installation failure (for whatever reason), latest after 2 hours without update from the application the Operator can assume that the activation / installation failed and react accordingly.

Linked Work Items	<p>is derived from: SPT2TS-49111 - An update for CCS on-board components is in operational mode within 2 weeks</p> <p>is derived from: SPT2TS-1846 - CCS configuration activation [e]</p> <p>has parent: SPT2TS-122384 - Non-functional requirements</p>
-------------------	--

SPT2TS-124471 - All building block interfaces are 'future proof' and backwards compatible.

All interfaces of building blocks of the CCS system shall be designed and implemented as 'future proof' and backwards compatible. This has to be considered when evaluating different technical solutions.

Linked Work Items	<p>is derived from: SPT2TS-124459 - Define backwards compatible and 'future proof' interfaces</p> <p>is derived from: SPT2TS-1845 - CCS configuration distribution [d]</p>
-------------------	--

	has parent: SPT2TS-122384 - Non-functional requirements _ is implemented by: SPT2TS-124929 - BuildingBlockConfiguration "configuration.json" document
--	--

5 Status of the work, open points, issues

Mid level concept description for the configuration management

In order to define the System Requirements of the single components with regard to the CCS configuration management process we need to develop a Logical Concept describing the agreed design guidelines for the configuration management process, the required content of the different standardised manifest files, the required content of the standardised distribution job files, and how the different roles are responsible for generating the different files. This Logical Concept is the basis to define the System Requirements of the single components of the CCS system with regard to the CCS configuration management process.

The document "Logical Concept" has been generated and reviewed.

SPT2TS-122392 - System @ Operator: synchronisation between TMS and update activation for specific CCS building blocks

It has been recognised as an open point how the synchronisation between Traffic Management System (TMS) and the update activation point in time for specific CCS building blocks should occur. The following variants have been identified:

- The TMS user and the user defining the distribution job for trackside components do synchronise this information between each other: from person to person (not specifically defining how this occurs).
- The information exchange is solved at the level of the technical systems: TMS application gets informed by the trackside system application (infrastructure) generating the distribution jobs that specific CCS building blocks will be out of service (performing the update) at a given point in time (according to the definition in the distribution job).

The information exchange allows at TMS level to consider in the planning activities that specific routes will not be in operation at a given point in time.

Note: The CCS on-board is not affected by this issue.

6 Tables

Please update the table of figures.

Please update the table of figures.