System Architecture

Abstract	This document describes the System Architecture of the Computing Environment (CE) and specifies the Computing Platform as execution environment for railway Functional Systems. It describes the architecture, principles, subsystems, interfaces, and operational concepts required to deploy, operate, maintain, and recover Functional Systems on virtualized Commercial-Off-The-Shelf (COTS) hardware.
Config Item	System Architecture Description
Document ID	30 Deliverables/System Architecture#910470 System Architecture
Classification	Public
Status	Open
Version	
Revision	910470
Last Change Date	2026-06-12

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

Approval by reviewers (at end of 'In Review by System Pillar")

Status	Open
Type of Approval	Document Review
Attachments	

Approval by Approvers (at end of 'In Approval by System Pillar")

Status	Open
Type of Approval	Document Approval
Attachments	

Table of Content

1 Preamble

1.1 Purpose

SPT2CE-3524 - The overall focus of the system architecture document is on the Computing Platform and its interfaces (I2 and I3). Interface I1 covering the configuration management, diagnostic and security aspects will be covered in more detail in a later version of the system architecture document.

Note: Please refer to previous SP CE documents for background Links are provided in the reference section. [Open **]**

1.2 Intended Audience

SPT2CE-3525 - This document is intended for the key players shaping the future of rail, specifically **Infrastructure Managers**, **Railway Undertakings**, **CCS Suppliers**, and **Sector Organisations**. It should be noted, however, that **this list is not limited to the mentioned audience**; the content serves as a valuable resource for any professional, regulator, or partner working within the broader rail and transport ecosystem. **[** Open **]**

1.3 Document Context

SPT2CE-3446 - This document contains the system architecture of computing environment and closes the gaps of the previous document 'System Analysis' by defining the Computing Platform subsystem and its interfaces I2 and I3. **[** Open **]**

1.4 Glossary

Term (Abbreviation) **Description**	**Referenced**
Application Execution Environment (AEE) The Application Execution Environment refers to the combination of Runtime Environment and Safety Environment. The safety environment is excluded for the basic integrity applications	here...
BBC (DR) (BBC (DR)) Building Block Configuration which contains the deployment rules for the Functional System (FS).	here...
BBC (InSW) (BBC (InSW)) Building Block Configuration which contains the initial software for the Initial Functional System (FS) Compartment.	here...
Building Block Configuration (BBC) Data for a Building Block, see Configuration Update Concept of Transversal.	here...
Commercial-of-the Shelf (COTS) Components available for public purchase without modification, listed in manufacturer's standard catalog (Hardware/Software)	here...
Compartment (Comp) A Compartment is a consistent, integrated entity which is part of a FS. It can be deployed on either a Physical or a Virtual Computing Element.	here...
Compartment Execution Environment (CEE) The Compartment Execution Environment refers to the combination of Physical Computing Element and Virtualisation Environment.	here...
Computing Environment (CEnv) A computing environment encompasses the hardware, software, network resources, and services that enable the deployment, operation, and management of applications or services. Computing environment includes the application execution environment and the computing platform.	here...
Computing Platform (CP) The Computing Platform provides and manages computing resources and communication resources for functional systems (specialised IO are not included). It contains CP hardware (physical computing element(s) and communication hardware) and CP software (virtualisation environment and platform management). Note: The CP shows an abstract view and may contains several Physical Computing Elements (PCE)s.	here...
Computing Platform Hardware (CPHW) Physical Computing Elements and network hardware.	here...

Term (Abbreviation) **Description**	**Referenced**
Computing Platform Software (CPSW) Provides platform-level services (e.g., resource allocation, compartment execution environment, and platform management).	here...
External Diagnostic, Configuration and IT Security Interface(s) (I1) The external Diagnostic, Configuration and IT Security Interface I1 (Interface 1) comprises communication-based interfaces between rail systems and central infrastructure components such as diagnostics, IT-security services and remote update. Note: This interface is implemented through SMI, SDI and SSI	here...
Functional System (FS) A Functional System is a comprehensive set of self-contained Compartments, assumed to be provided as one product by a single vendor. Depending on its overall function, it has a specific SIL assigned.	here...
Functional System Deployment Rules (FSDR) The Functional System Deployment Rules comprises all necessary information for deploying the respective Functional System onto specific approved Compartment Execution Environment(s). These deployment rules are compiled as part of the FS integration process and are part of each integrated, tested and qualified/approved Functional System along with its FS Compartments and all necessary approval documentation.	here...
Hardware Compatibility Interface (I2) The Hardware Compatibility Interface I2 (Interface 2) provides the compatibility requirements for the specific hardware used below, enabling easy replaceability of commercial of-the-shelf hardware procurable from a well-sized market of hardware vendors. Note: This is not really an interface, but rather a compatibility list of allowed hardware incl. CPU, memory, etc.	here...
Native Hardware Access (NHA) Specific software component(s) providing the additional functions needed by the Functional Systems (especially the Safety Layer) that are not available in COTS solutions.	here...
Network Hardware (NW HW) The collection of network hardware (e.g., switches) used to connect the PCEs. The network topology is not detailed in the example, as it may be specific to the concrete Computing Platform implementation.	here...
Operational Interface (I0) The I0 is the sum of all operational interfaces used from Functional Systems (as eg. an RBC) to communicate with other Functional Systems (as eg. an IXL). Examples for these set of interfaces are the Eulynx Interfaces (SCI-xx) or interfaces like Euroradio or TSI-standardised interfaces.	here...
Physical Computing Element (PCE) The Physical Computing Element refers to the physical device (e.g. Server) providing compute resources.	here...

Term (Abbreviation) **Description**	**Referenced**
Platform Management (PM) The platform management manages the computing platform resources and is a part of the Computing Platform software.	here...
Platform Management Adapter (PMA) This software component provides a connection between COTS platform management and railway specific standardised interfaces (SMI, SDI and SSI)	here...
Runtime Environment (RTE) The Runtime Environment refers to the software needed to provide the services of the Runtime Layer in a single Compartment.	here...
Safety Layer (SL) The Safety Layer implements all the technical safety principles related to fulfilling the requirements of EN 50126, EN 50716 (formerly 50128), EN 50129, EN 50159 (e.g., composite fail safety, fault tolerance, voting mechanisms, redundancy mechanisms for availability, safety communication layers etc.) that are needed to enable the execution of Functional Applications up to SIL4.	here...
Standard Diagnostic Interface (SDI) Standard Diagnostic Interface as defined by EULYNX / System Pillar	here...
Standard Maintenance Interface (SMI) Standard Maintenance Interface as defined by EULYNX / System Pillar	here...
Standard Security Interface (SSI) Standard Security Interface as defined by EULYNX / System Pillar	here...
VCE Management (VCE Management) Virtual computing element manages the compartment. VCE management offers functions to create, configure, and remove compartments. VCE management has its own processes, tools and policies that shape its functions and enable the control and operation of the compartments.	here...
Virtual Computing Element (VCE) The Virtual Computing Element refers to virtually provided compute resources with computing resource guarantees.	here...
Virtualisation Environment (VE) The Virtualisation Environment contains all software needed to provide (multiple) Virtual Computing Elements on a single Physical Computing Element.	here...
Virtualisation Interface (I3) The Virtualisation Interface I3 (Interface 3) is used to provide a standardised interface above the virtualisation layer so that applications or higher platform layers are independent of a specific implementation of the computing hardware.	here...

2 Overview

2.1 Overall Description

SPT2CE-3352 - The following figure shows the Computing Platform and its surrounding actors.

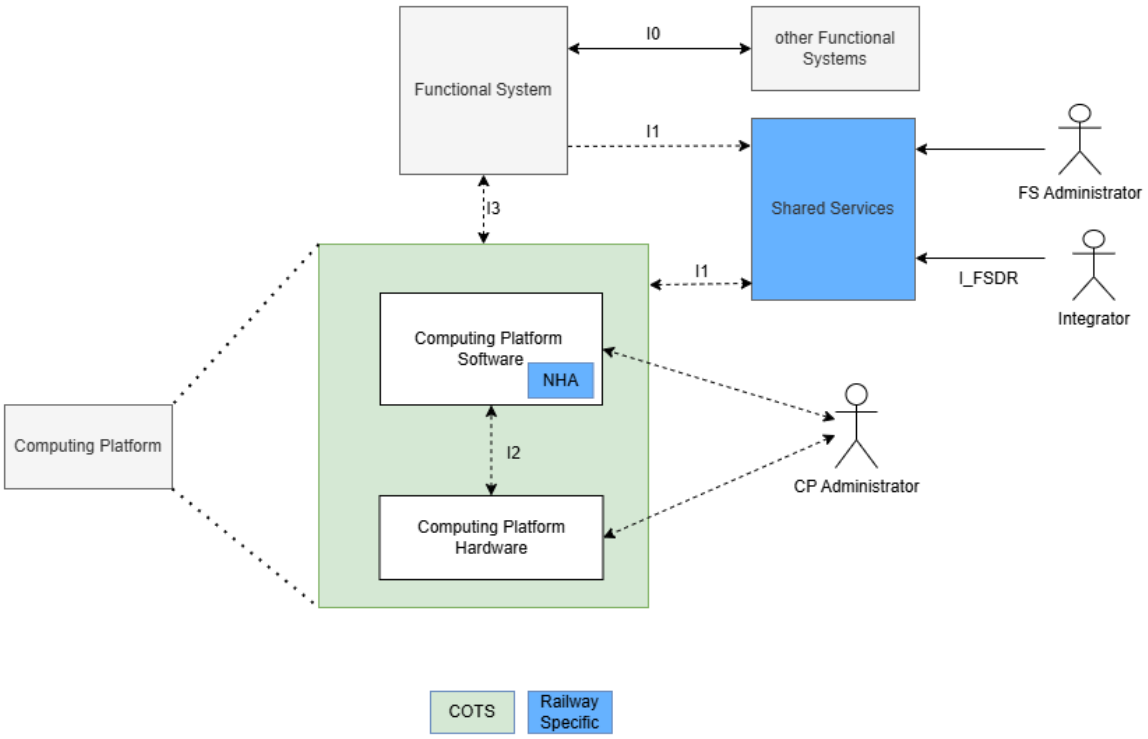


Figure 1 :Computing Platform Architecture

The following tables describe the element of Figure 1.

Element	Description
Computing Platform	System under consideration.
Computing Platform Hardware	All hardware components which are part of the Computing Platform.
Computing Platform Software	All software components which are part of the Computing Platform.
Functional System	Operational software running on the computing platform. This software provides the railway function.
Other Functional System	Other Functional System which has a communication relation with a Functional System running on the Computing Platform. It might be an external system (e.g. object controller, legacy system) or a Functional System running on the same or another Computing Platform.
Shared Services	Shared services provide central functionality.

Element	Description
FS Administrator	Functional System Administrator- entity responsible for operating the railway system is, as such, responsible for the operation of the functional systems.
CP Administrator	Computing Platform Administrator- The entity responsible for operating the Computing Platform provides services to the entity responsible for running the Functional Systems.
Integrator	The entity responsible for integrating the FS onto the CP according to the FSDR.

Abbreviation	Interface	Description
I0	Operational interfaces	Interfaces between Functional Systems. Communication over these interfaces is IP-based.
I1	External Diagnostic, Configuration, and IT Security Interface(s)	Interfaces to central infrastructure components. Communication over these interfaces is IP-based.
I2	Hardware Compatibility Interface	Interface between Computing Platform Software and Computing Platform Hardware
I3	Virtualisation Interface	Interface of the Computing Platform to provide services to the Functional Systems.
I_FSDR	Logical interface	Sum of configuration files for the deployment of FS onto the computing platform

[Open **]**

SPT2CE-3353 - The following figure shows the sub-components of the Computing Platform Hardware.

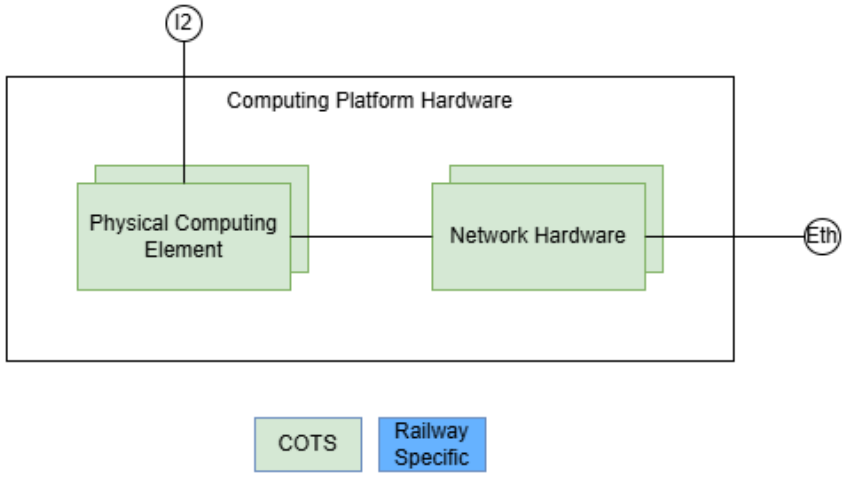


Figure 2 Computing Platform Hardware

All hardware components used for the Computing Platform could be standard COTS products conforming to the environmental requirements of the installation location.

Hardware Component	Description
Physical Computing Element	Computing hardware that executes the software, e.g. servers.
Network Hardware	Hardware components needed to connect the PCEs for communication, e.g. network switches.

Abbreviation	Interface	Description
Eth	External communication interface	Physical communication interface used for communication with external partners (e.g. shared services, other).
I2	Hardware Compatibility Interface	Compatibility interface between Computing Platform Hardware and Computing Platform Software.

[Open]

SPT2CE-3354 - The following figure shows the sub-components of the Computing Platform Software.

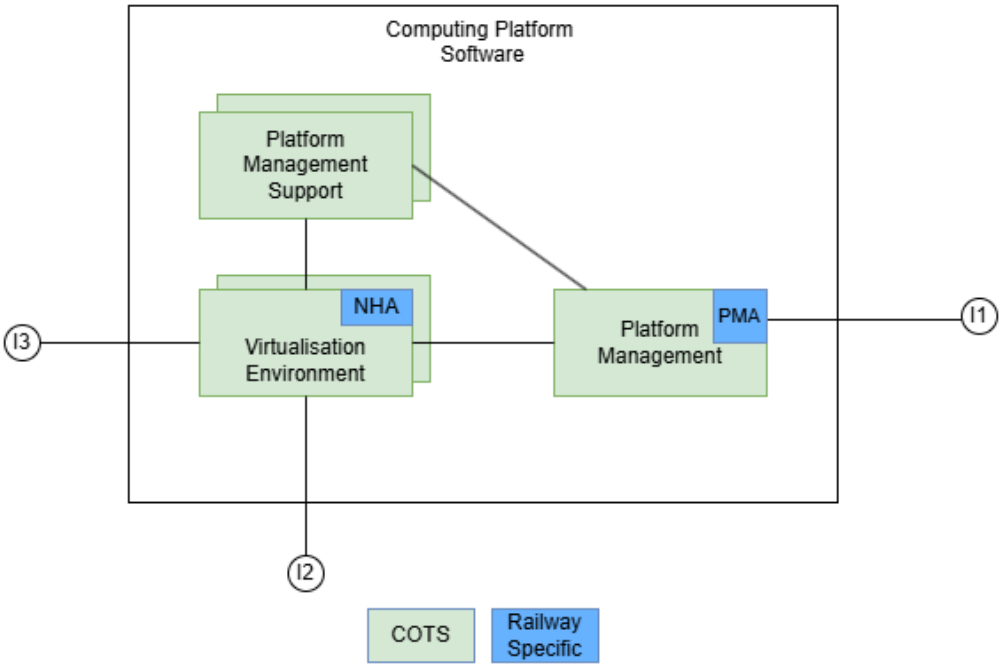


Figure 3 Computing Platform Software

Software Component	Description
Virtualisation Environment	General purpose COTS software component(s) running on each PCE to provide the general purpose COTS functions of the Computing Platform regarding I3 (e.g., hypervisor, communication drivers, storage drivers, ...).
NHA	Native Hardware Access- Specific software component(s) providing the additional functions needed by the Functional Systems (especially the Safety Layers) that are not available in general purpose COTS solutions.

Software Component	Description
Platform Management	Software component(s) that provide the central management functions for the Computing Platform. They might run as a compartment on VEs of the Computing Platform or be independent of the VEs. The Platform Management is a combination of general-purpose COTS components along with railway-specific components.
PMA	Platform Management Adapter- This software component provides a connection between general purpose COTS platform management and railway specific standardised interfaces (SMI, SDI and SSI)
Platform Management Support	An optional support component for the Platform Management that can run in parallel to FS compartments on a VE instance.

[Open **]**

SPT2CE-3355 - The following figure shows an example deployment of the components of the Computing Platform and 2 exemplary Functional Systems that are organised as 2oo3

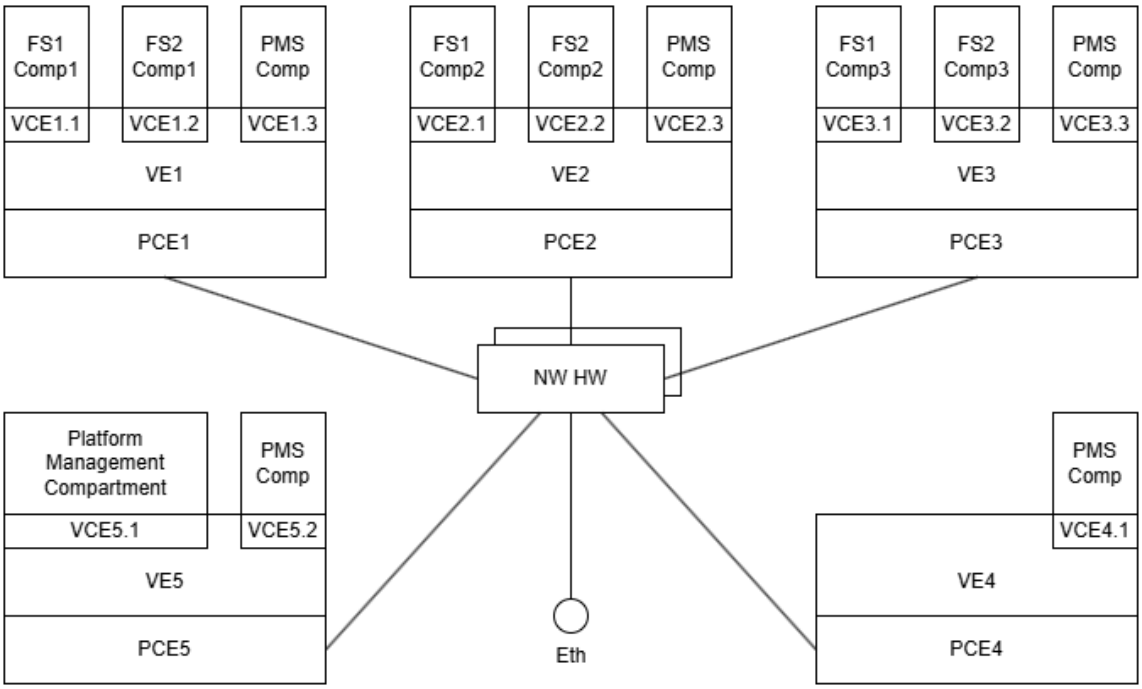


Figure 4 Example deployment on Computing Platform

The following table describe the elements in the figure 4.

Element Abbreviation	Element	Description
PCE<n>	Physical Computing Elements	The example consists of 5 PCEs (e.g., servers). Three servers are used to run the functional systems (PCE1 through 3), one is a spare (PCE4), and one is used to run Platform Management (PCE5).

Element Abbreviation	Element	Description
VE<n>	Virtualisation Environment	On each of the 5 PCEs, a VE (e.g., a hypervisor) runs.
VCE<n>.<m>	Virtual Computing Element	For each Compartment running on the Computing Platform, the Platform Management creates a VCE in conjunction with the VEs (e.g., the definition of the virtual machines in the hypervisor).
NW HW	Network Hardware	The collection of network hardware (e.g., switches) used to connect the PCEs. The network topology is not detailed in the example, as it may be specific to the concrete Computing Platform implementation.
FS<n> Comp	Functional System Compartment	The example consists of 2 Functional Systems (FS1 and FS2), where each has 3 compartments running on different PCEs (Comp1, Comp2, Comp3).
PMS Comp	Platform Management Support Compartment	In the example, on each PCE, one compartment runs and provides support functions for Platform Management (e.g., monitoring functions not implemented in the hypervisor). How such functions are realised, or whether they are even needed, is specific to the concrete Computing Platform implementation.
PM Comp	Platform Management Compartment	In the example, the central Platform Management functions are implemented as a Compartment running in the same environment as the Functional System compartments. How the Platform Management functions are realised, or if they run in a different environment, is specific to the concrete Computing Platform implementation. The availability requirements are defined for specific deployment.
Eth	Physical interface to external systems	The whole system, including the Computing Platform and the Functional Systems, has communication interfaces to the Shared Services and other Functional Systems that are not running on this Computing Platform.

[Open **]**

SPT2CE-3356 - Note: Computing platform software is assumed to be mainly general purpose COTS with some railway specific non-COTS components such as NHA and Platform Management Adapter. **[** Open **]**

2.2 Architecture Principles

2.2.1 Identifiers

SPT2CE-3569 - CP-related identifiers for orchestration of the CP software on the CP hardware

- **pceID** for the physical computing element, e.g., pce1
- **veID** for the instance of the virtualization environment, e.g., ve1 running on pce1
- **vceID** for the virtual computing element on the virtualization environment, e.g., vce1.1 on ve1 on pce1

[Open **]**

SPT2CE-3560 - Building Block Identifiers (bbIDs) for the configuration process with SFC

Each individual resource involved in the configuration process via interface I1 SMI with SFC is a building block with its own unique **bbID**.
The following bbIDs are required:

- **One bbID for the PM**, e.g., bbPM-1
  Needed to load BBCs via SMI from SFC into the PM in the context of “FS deploy”, see figure below “Deploy”.
  This bbID for the PM is set by the CP-Admin within the PM context for the installation PM.

- **Individual bbID for each FS compartment,** e.g., bbFSA-C1, bbFSA-C2, and bbFS-C3 for the three compartments of FS A.
  An FS defines the individual bbID for each FS compartment it belongs.
  The bbID of an FS compartment is set during the creation of the FS compartment in the context of the deployment of the initial software.
  If a FS compartment moves onto another VCE (in context of recovery of a failure) the bbID also moves together with the FS comp software.

[Open **]**

SPT2CE-3559 - Building Block Configuration Identifiers (bbcIDs) for the configuration process with SFC

Each BB contains one or more BBCs with belonging **bbcID**s.
The following bbcIDs are required per FS compartment:

- bbcID for the **initial SW**, e.g., bbcFSA-C1-IN, bbcFSA-C2-IN, bbcFSA-C3-IN
- bbcID for **deploy rules**, e.g,. bbcFSA-C1-DR, bbcFSA-C2-DR, bbcFSA-C3-DR
- bbcID for **functional SW**, the number of BBCs depends on the details of the FS solution.
  e.g. bbcFSA-C1-SWx, bbcFSA-C2-SWx, bbcFSA-C3-SWx

The FS compartment-related BBCs with initial software and deployment rules shall be packaged into one FS-related bbcFSA-DR to ensure that all needed FS data is available for the PM to decide on the selection of CP resources for all FS compartments.
In case of individual transfer of bbcFSA-C1/2/3-IN and bbcFSA-C1/2/3 DR several transfer steps would be necessary with the potential for interruption of the process and inconsistent versions of the BBCs within the PM..

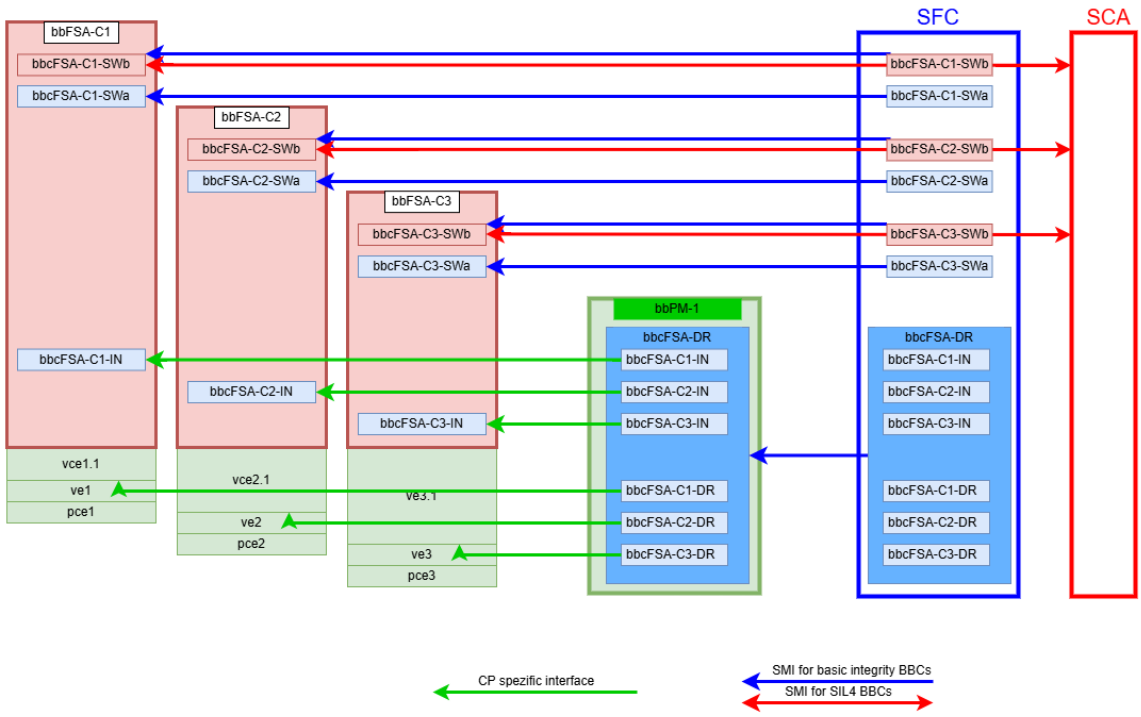


Figure 5 bbIDs and bbcIDs

For further details, see SPT2CE-3091 - Deployment of an FS

[Open **]**

SPT2CE-3558 - Identifier for the FS as Subsystem

All FS compartments of an FS belong to the same Functional System, meaning they use the same **subsystemID**.

[Open **]**

SPT2CE-3590 - Basic Data Identifier

Each FS compartment has its own "**Basic Data Identifier**" (e.g. textfile)

- within the initial software, needed for starting the SMI process in the context of the deployment

and

- within the SIL4 data as part of the SIL4 bbc, needed for the safety related SMI process.

The Basic Data Identifier contains
- the bbID of the belonging FS Compartment
- the subsystemID of the belonging subsystem

[Open **]**

SPT2CE-3589 - CP related Identifiers

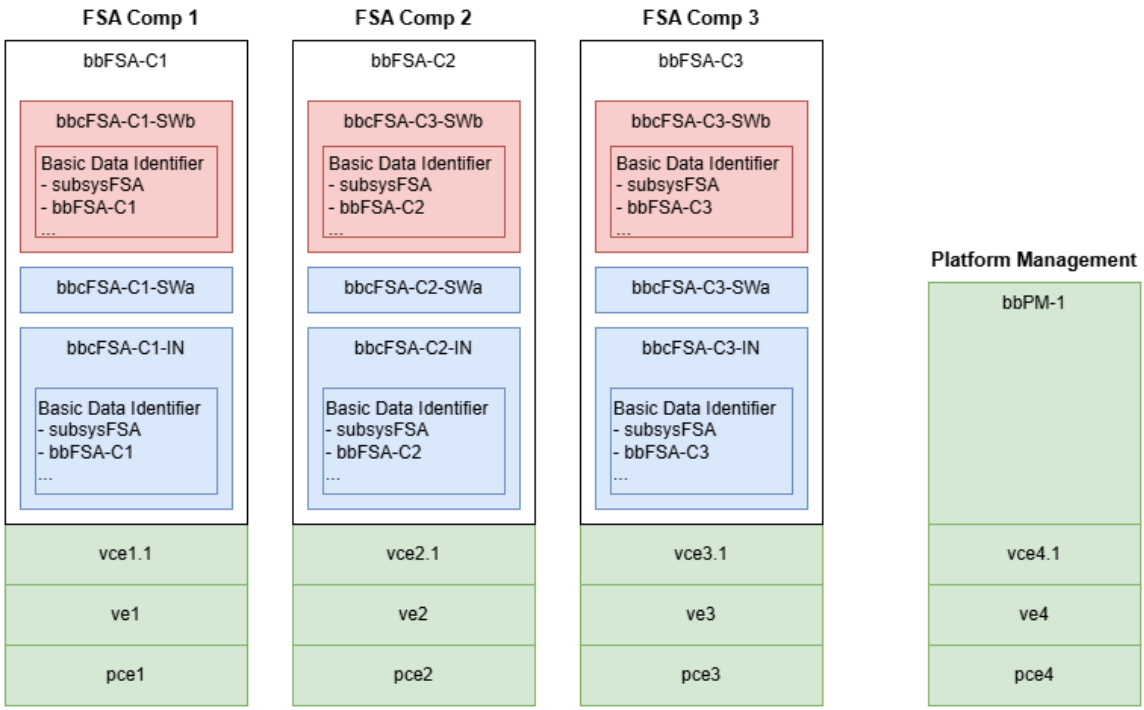


Figure 6 IDs in the layered architecture

[Open **]**

2.2.2 Diagnostic

SPT2CE-3588 - FS provides the product data model (product-specific)

- We provide the equipment data model for VCE **[** Open **]**

SPT2CE-3587 - Data Flow Diagnostic

- **HW Diagnostic**
  is installed on each CP hardware and provides the detailed state of the CP hardware (Diag HW-x) to the PM.
  Protocol defined by the HW diagnostic software.These protocols are assumed to be COTS.
- **Network Management**
  provides the detailed network state (Diag network) to the PM.
  The protocol is defined by the Network Management System. These protocols are assumed to be COTS.
- **Virtualisation Environment**
  provides
  - states of the individual VCEs (Diag VCE-x)
  - states of the VE itself (Diag VE-x)
  to the PM.

The protocol is defined by the VE solution. These protocols are assumed to be COTS.

- **FS Compartment**
  provides
  - state of the FS application (Diag FS-App)
  - state of the FS compartment (Diag FS-Comp)
  to the Shared Services Diagnostics.
  Protocol defined by Shared Services Diagnostic SDI.
  Diag FS-App is provided by each individual FS Compartment for redundancy.
- **Platform Management**
  aggregates the individual states (VCE, Hardware, Network) to PM and provides this information to the Shared Service Diagnostic. For aggregating network states into CEE states, the communication dependencies between FS components and external systems are defined in the FS deployment rules (DR).
  Protocol defined by Shared Services Diagnostic SDI.

The figure below shows the data flow exemplary for a SIL4 FS-A running in three FS compartments and a FS-B running in one FS compartment.

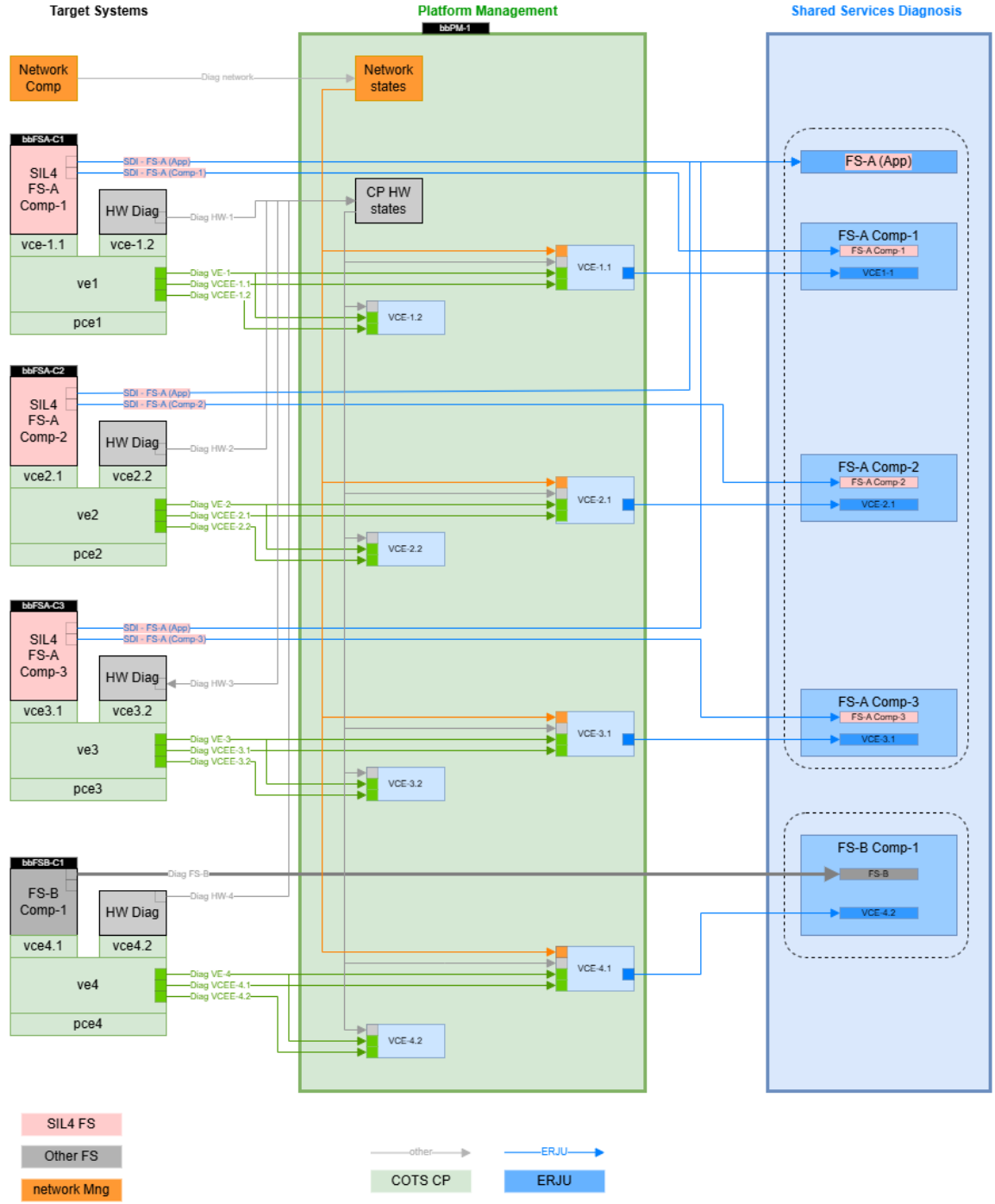


Figure 7 Data Flow Diagnostic

[Open]

2.2.2.1 Diagnostic Interface

SPT2CE-3598 - The following needs to be specified:

- The CP-related diagnostic data and protocols for provision to Shared Services Diagnostics. In this regard, standard protocols versus SDI must be considered.
- The necessary extensions within the CPs Platform Management for the evaluation of diagnostic data for the FS Compartment for the provision of FS Comp-related diagnostic data to Shared Services Diagnostics, in accordance with I1-SDI.
- In this context, it must be determined whether root-cause analysis for the detection of fault causes within the CP is performed by the CP's Platform Management or by Shared Services Diagnostics.

[Open **]**

SPT2CE-3597 -

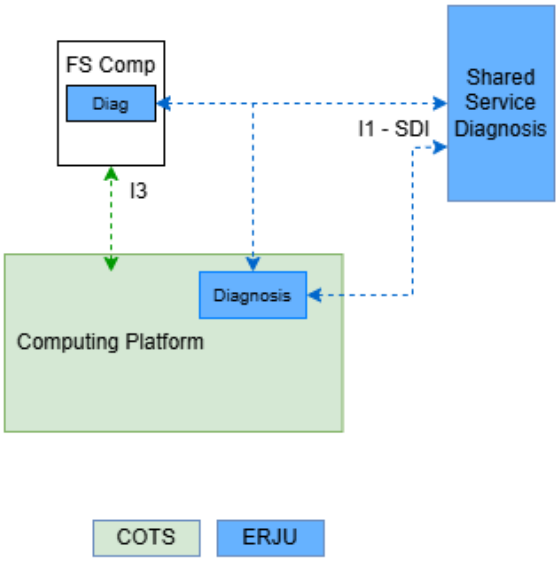


Figure 8 Diagnostic architecture

[Open **]**

SPT2CE-3606 - - Service function diagnostic (collector and aggregator) Protocol converter
- Product group set model (information model)

VMs location need to be provided by CE

Equipment and Product model

Each FS compartment reports himself ? or a collector collects all the data and combine the information at the source? **[** Open **]**

2.2.2.1.1 Diagnostic CP SW

SPT2CE-3605 - The allocation of the diagnostic functionality to monitor the CP software has to be evaluated regarding the the dependency to the interfaces I1 and I3.
SPT2CE-2322 - Fct-CP SDI - Provide CP diagnostic data according I1-Diag SDI **[** Open **]**

2.2.2.1.2 Diagnostic FS Comp

SPT2CE-3604 - The allocation of the root cause analysis for all failures affecting the FS compartment runtime has to be evaluated regarding the the dependency to the interfaces I1 and I3. **[** Open **]**

2.2.2.2 Maintenance

SPT2CE-3603 - The allocation of maintenance related functions to automate the recovery of failures within the CP software, CP hardware or FS Compartment has to be evaluated regarding dependency to interface I3 and I2. **[** Open **]**

2.2.2.3 I1 - Diagnosis (SDI,..)

SPT2CE-3602 - - Basic diagnosis data models (Compartment OPC-UA model)
own running state (of the FS compartment)
state of comm-connections to neighbor FS compartments

- COTS Diagnosis protocols <> SDI (OPC UA) - to be supported by Shared Services Diag ?

SPT2CE-2593 - Open #SDI - Identification of FS Comp failure by CP ?
Result of discussion Transversal unclear.
Questioned is the data source (not the data sink).
Standardised I1 SDI information (kind of "heartbeat") from the FS Compartment to be processed by the CP ?
Where to place the root cause analysis for failures of FS compartments (failure in COTS HW / COTS SW / network / FS Comp) - within CP or on side of Shared Services Diag ?
SPT2CE-2744 - Open #SDI-Allocation of aggregation of diagnostic data for FS
Aggregation of FS compartment related Diag-data to FS Diag-data - within CP or on side of Shared Services Diag ?
SPT2CE-2557 - Open #SDI - Process and scenario for stop of a FS by Shared Services Diagnostics ?
Result of discussion Transversal unclear.
Does SMI provide functions for start / stop of FS systems ?
Needs to be processed for all belonging FS compartments in parallel.
[Open **]**

2.2.3 Configuration Management

SPT2CE-3594 - Version consistency across the FS compartments

Each FS Compartment consists of one or more BBCs, depending on safety relevance and, in detail, on the specific solution.
SIL4 BBCs must be version consistent across all FS Compartments of a SIL4 FS. It's the responsibility of the safety layer of the SIL4 FS to ensure this consistency.
For basic integrity BBCs, it depends on the FS if a cross dependency between neighbor compartments exists and in which way a version consistency is needed

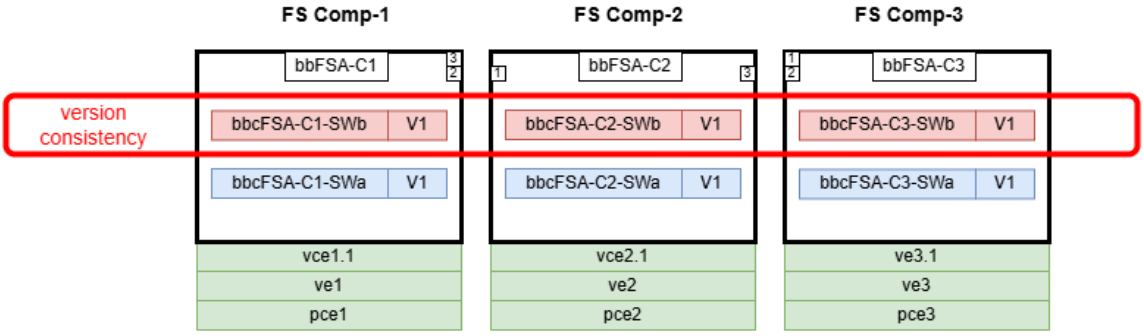


Figure 9 SIL4 and basic integrity BBCs of a SIL4 FS-A with 2oo3 principle

[Open **]**

2.2.3.1 Deployment of an FS

SPT2CE-3593 - Deployment of a FS

The deployment of an FS is done in basically three steps:

1. The FS Admin initiates loading the bbcFSA-DR with the deploy rules and the initial FS-A software from SFC to the Platform Management.
    a. The platform management needs the complete deployment rules (for all 3 FS Comp) to decide on the available HW (Servers, Network) resources.
    b. The platform management stores the data (deploy rules, BBC(In)) for use in later recovery scenarios.
    c. The stored FS data (deploy rules and initial SW) is not safety related. It does not include any functionality in the sense of “application logic”.
2. The CP admin selects the prepared PCEs (pce1 with ve1, pce2 with ve2, pce3 with ve3) and initiates the configuration of the network
3. The PM evaluates the deployment rules for FS Comp-1 bbcFSA-C1-DR, configures ve1 and pce1, and creates the initial FS Compartment for bbFSA-C1 with the initial software bbcFSA-C1-IN within pce1.1.
4. The initial FS Comp-1 = bbFSA-C1 in vce1.1 has a maintenance port to interface with the SFC and uses it to initiate the upload of the functional software as bbcFSA-C1-SWa and bbcFSA-C1-SWb via "pull" via a reverse connect from the SFC into bbFSA-C1.

BBCs are based on the dependency tree within SFC.

For SIL4 BBC, the SCA is involved in safety-related processes to confirm the correct BBC activation.

5. as 3. for FS Comp-2 = bbFSA-C2 within vce2.1 on ve2 on pce2
6. as 4. for FS Comp-2

   FS is now running with FS Comp-1 and FS Comp-2 as 2oo2
7. as 3. for FS Comp-3 = bbFSA-C3 within vce3.1 on ve3 on pce3.
8. as 4. for FS Comp-3

   FS is now running with FS Comp-1, FS Comp-2, and FS Comp-3 as 2oo3.

The figure below shows the data flow for deploying exemplarily for an FS-A running with the 2oo3 principle.

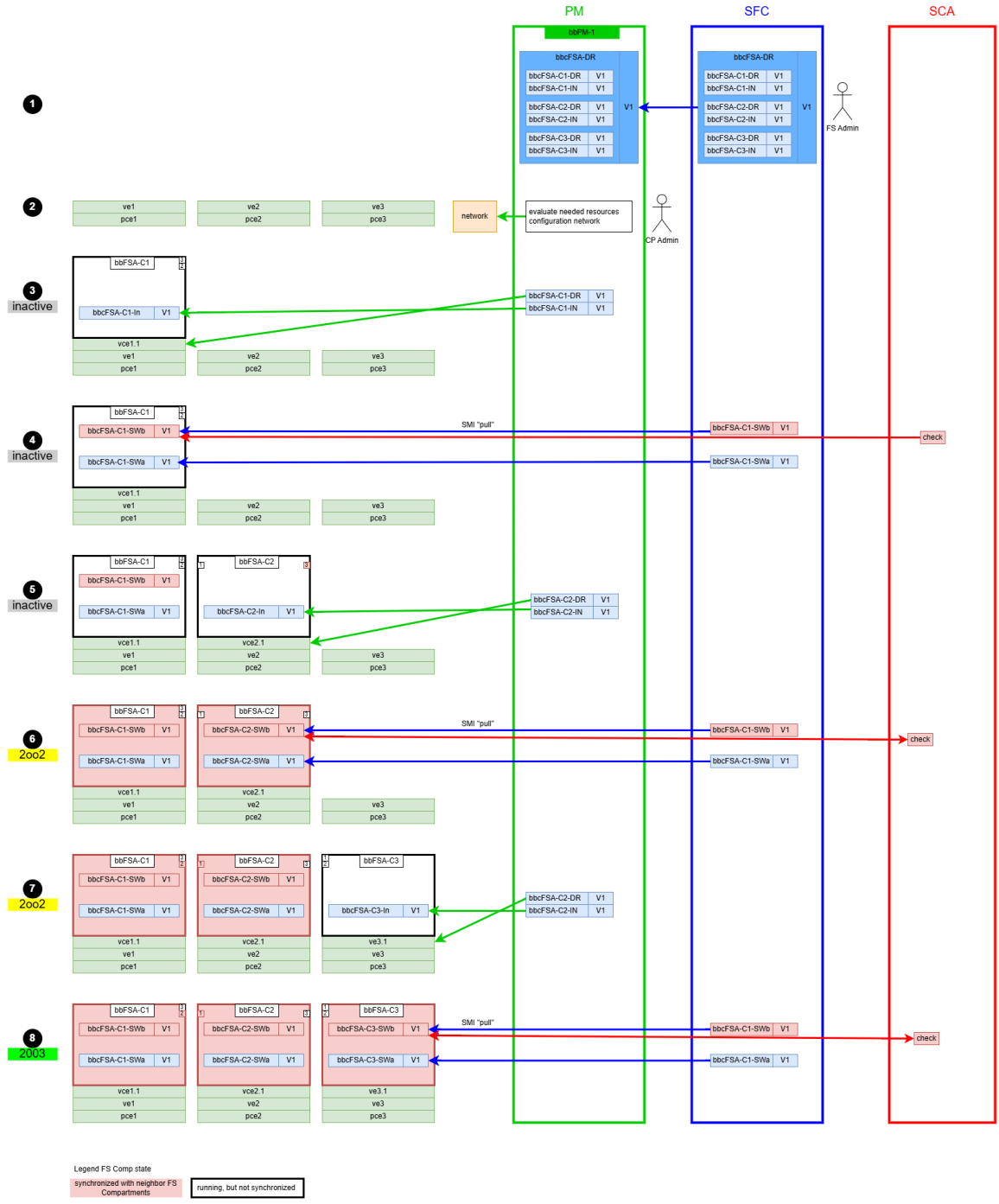


Figure 10 Software Deploy for a SIL4 FS with 2oo3 principle

[Open]

2.2.3.2 Update

2.2.3.2.1 Update CP SW

SPT2CE-3592 - Update CP SW

CP SW updates are performed by each PCE instance. For this, the FS redundancy principle can be used by updating the compatible CP SW into the redundancy channel "one after the other."

The figure below shows the steps for updating a compatible CP SW onto CP HW-1.

Starting point: SIL4 FS is running with full availability (2oo3).

1. New CP SW version V2 is pre-loaded onto pce1.
   Initiated by the CP Admin.
2. VE version V2 is activated. This leads to stopp of FS Comp-1 = bbFSA-C1.
   FSA keeps running as 2oo2.
3. FS Comp-1 is restarting and synchronizing with the neighboring compartments.
   FSA achieves 2oo3 mode again.

There is no cross-dependency between neighbor compartments for the CP SW version, so it's possible to run the FS compartments with different versions of the non-SIL CP SW.

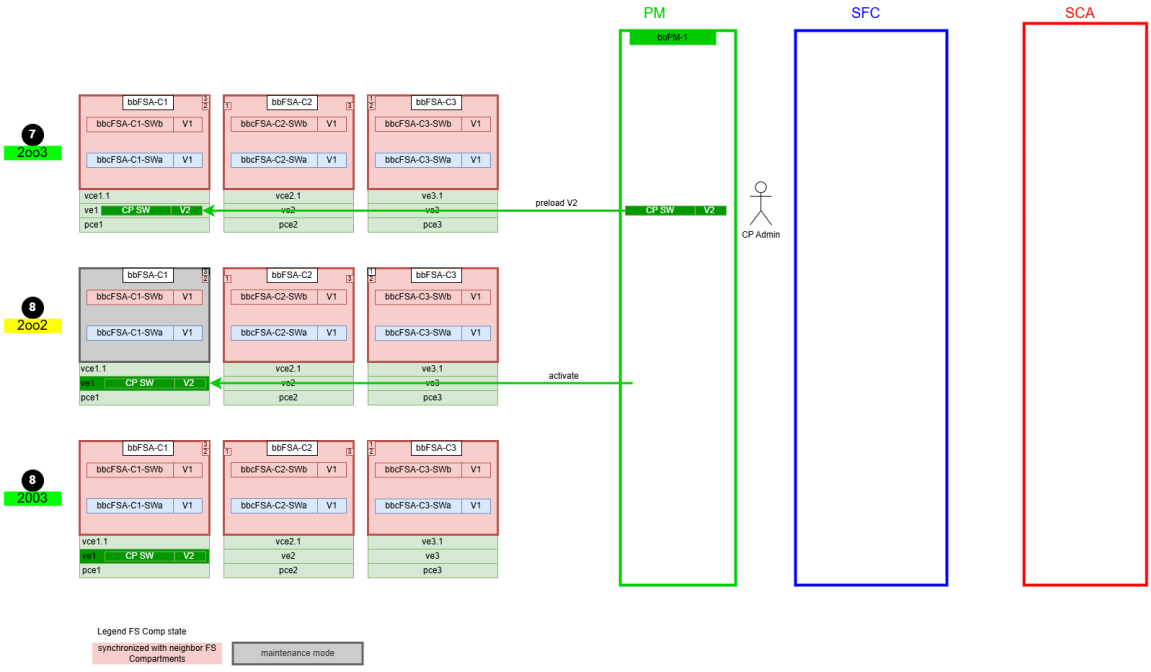


Figure 11 Update CP SW

[Open]

2.2.3.2.2 Update SIL4 FS BBCs

SPT2CE-3591 - The update of safety related BBCs from SFC into the FS compartments is done FS compartment-wise with involvement of the SCA. Each FS Compartment provides its own safety hash and receives the "confirmation" from the SCA.
Each FS Compartment gets its own operational token.
The SCA does not require knowledge of the interrelationships of the FS compartments.

It's the responsibility of the safety layer within the FS to

- ensure the overall SW version consistency of all BBC within the belonging FS compartments
- ensure that an individual FS compartment with a new BBC version does not process functional actions before it's successfully synchronized with a neighbor compartment.

The figure below shows the steps for a compartment-wise update of a SIL4 FS with a 2oo3 principle from BBC version V1 to V2.

1. Starting point: FS running as 2oo3

   .

   FS Compartment-wise preload (one after the other):
2. Preload of bbcFSA-C1-SWb version V2 into bbFSA-C1
3. Preload of bbcFSA-C2-SWb version V2 into bbFSA-C2
4. Preload of bbcFSA-C3-SWb version V2 into bbFSA-C3

   .
5. Deactivate all three FS compartments at the same time point by SCA
   All FS compartments are in maintenance mode
   FS is "inactive".

   .

   FS Compartment-wise activation of version V2:
6. Activate bbcFSA-C1-SWb version V2 with involvement SCA
7. Activate bbcFSA-C2-SWb version V2 with involvement SCA
   FS active in 2oo2 mode with version V2.
8. Activate bbcFSA-C3-SWb version V2 with involvement SCA
   FS active in 2oo3 mode with version V2.

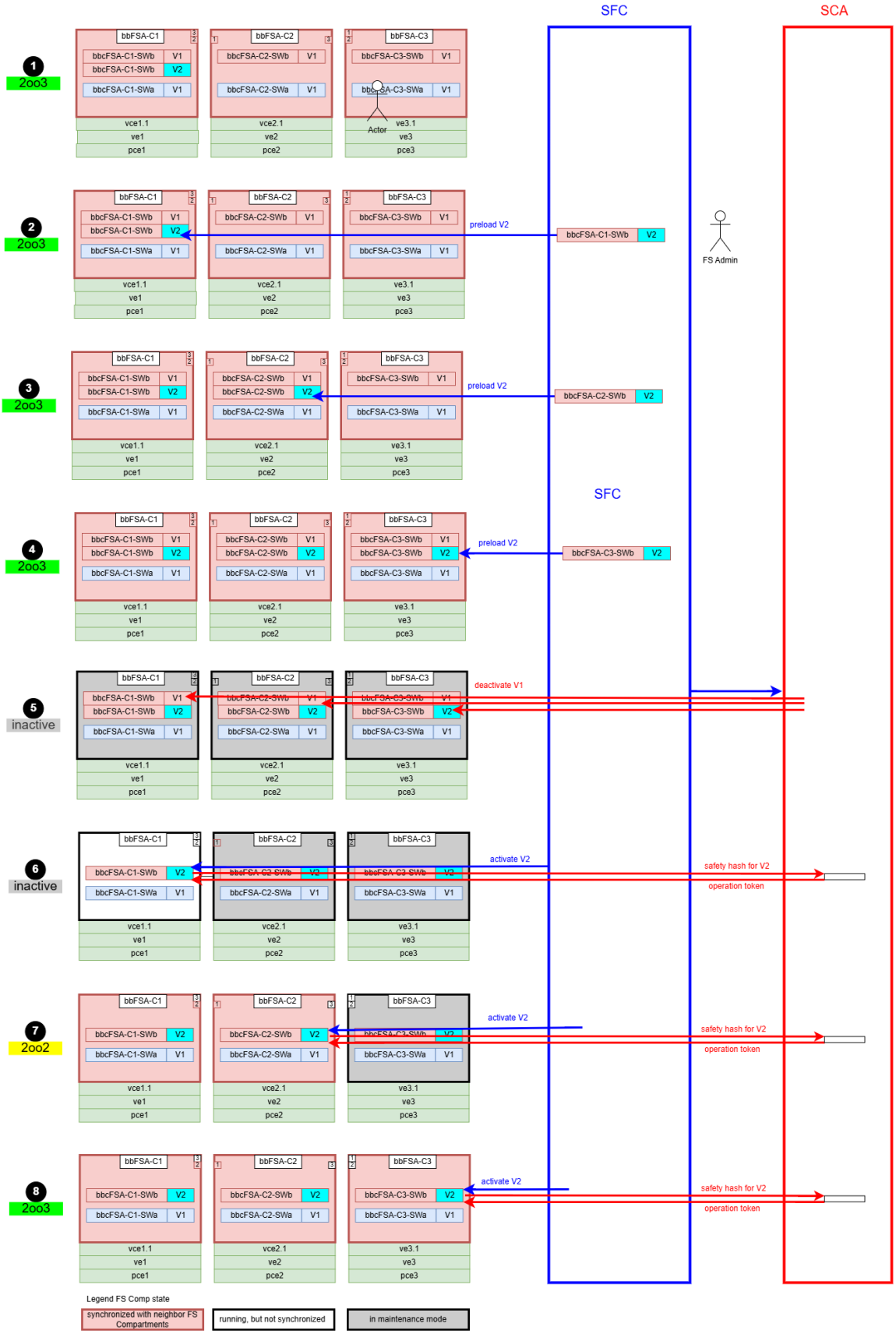

SFC
SCA
1
2oo3
2
2oo3
3
2oo3
4
2oo3
5
inactive
6
inactive
7
2oo2
8
2oo3
bbFSA-C1
bbFSA-C2
bbFSA-C3
bbcFSA-C1-SWb
bbcFSA-C2-SWb
bbcFSA-C3-SWb
bbcFSA-C1-SWa
bbcFSA-C2-SWa
bbcFSA-C3-SWa
V1
V2
vce1.1
vce2.1
ve3.1
ve1
ve2
ve3
pce1
pce2
pce3
Actor
FS Admin
preload V2
deactivate V1
activate V2
safety hash for V2
operation token
Legend FS Comp state
synchronized with neighbor FS Compartments
running, but not synchronized
in maintenance mode

Figure 12 Update FS with new version of SIL4 BBC

[Open **]**

2.2.3.2.3 Update nonSIL BBC of a SIL4 FS

SPT2CE-3586 - Update nonSIL BBC of a SIL4 FS

The update of non-safety-related BBCs of a SIL4 FS from SFC into the FS compartments is done FS compartment-wise.
In this case, "new non-SIL BBC is compatible with running SIL4 BBC," it's possible to perform these updates without stopping the running FS. For this the redundancy principle of the FS can be used by updating the compatible nonSIL BBC into the redundancy channel "one after the other".
The figure below shows the steps for updating a compatible non-SIL BBC in FS Comp-1 of a SIL4 FS with a 2oo3 principle.

1. Starting point: FS is running with version V1 and full availability (2oo3)
2. Preload of compatible bbcFSA-C1-SWa version V2 into FS-Comp-1 = bbFSA-C1
3. Deactivation of bbcFSA-C1-SWa version V1, activation of bbcFSA-C1-SWa version V2.
   FS Comp-1 is in maintenance mode; FS is in reduced availability (2oo2).
4. FS Comp-1 is starting up with bbc-FSA-C1-SWa version V2 and synchronizing with the neighboring compartments.
   There is no cross-dependency to neighbor-compartments for the nonSIL BBC version, which means it's possible to run the compartments with different versions of the nonSIL BBC:
   FS Comp-1 is running with bbcFSA-C1-SWA version V2 (new)
   FS Comp-2/3 are running with the bbcFSA-C2/3-SWa version V1 (previous).

Update of FS Comp-2 and Comp-3 is done in the same way.

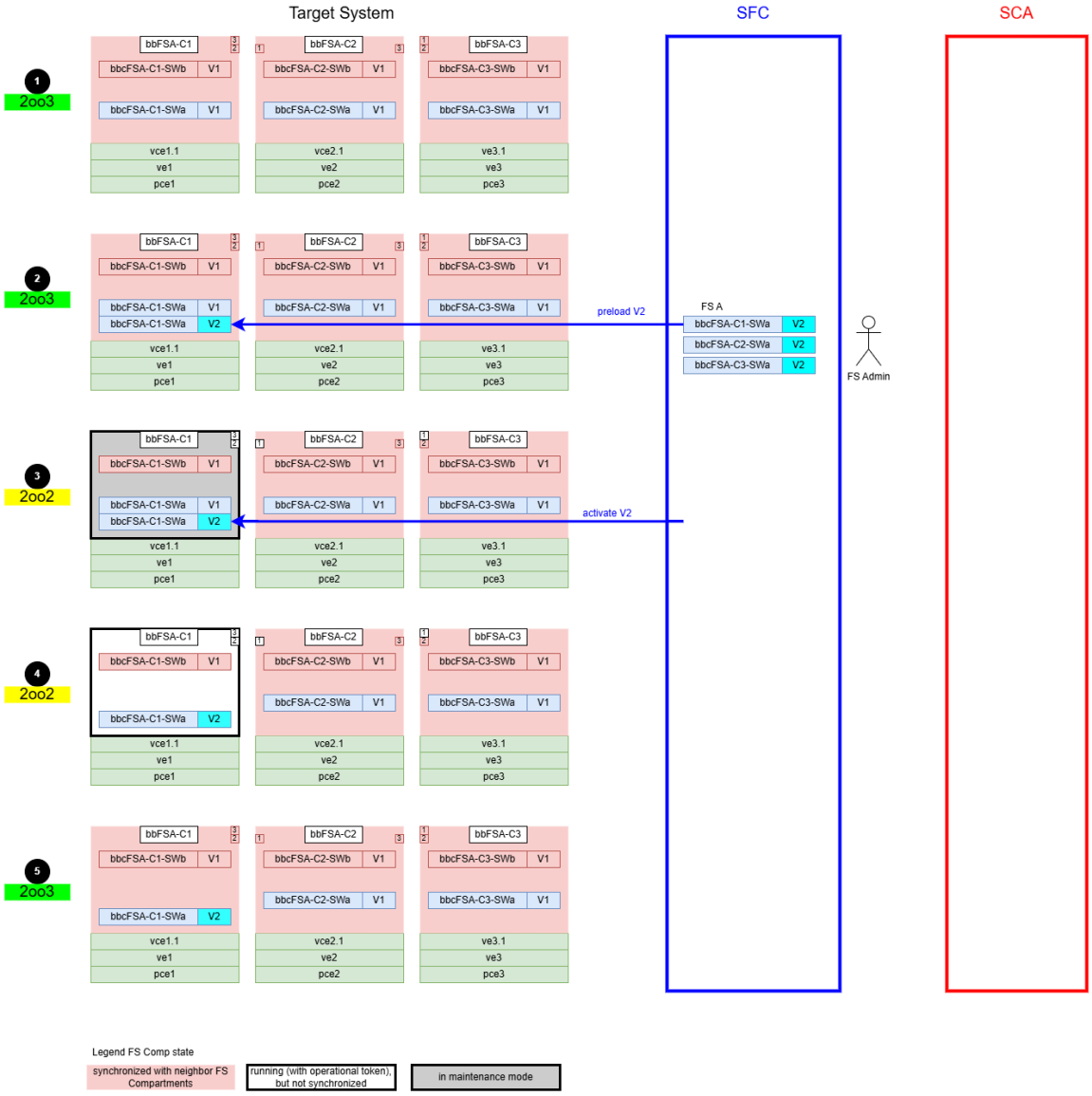


Figure 13 Update nonSIL BBC of a SIL4 FS

[Open **]**

2.2.3.3 Recovery

2.2.3.3.1 Recovery of an individual CP HW failure

SPT2CE-3582 - Recovery of Individual CP hardware failure.

The failure of a CP hardware is identified by the platform management. The maintainer is responsible for identifying and rectifying CP hardware failures.

The platform management is aware of the affected FS compartments installed on this failed CP hardware.

For the recovery of the affected FS compartment, the following steps are necessary:

1. Starting point: FS is running as 2oo3.
2. Failure of pce3 occurs and causes FS Comp-3 to stop.

FS Comp-1/2 identify the missing neighbor FS Comp-3. This leads to reduced FS availability to 2oo2 mode.
FS Comp-1 and FS-Comp-2 provide diagnosis data via interface I1-SDI to SFD.
PM provides diagnostic data via the I1-SDI interface to SFD.

3. The platform management evaluates the required recovery mechanism: the deployment of the FS Comp-3 software onto another PCE.
   The Platform Mngmt configures the network components according to the stored deploy rules of the FS compartment.
   Initiated by the CP Admin.
4. The Platform Mngmt uses the stored deploy rules bbcFSA-C3-DR and initial software bbcFSA-C3-IN and creates the initial FS Comp-3 as vce4.1 on pce4.
   PM provides diagnosis data for vce4.1 via interface I1-SDI to SFD.
5. All the functional software bbcFSA-C3-SWa and bbcFSA-C3-SWb is updated via I1-SMI into the FS Comp3 = bbFSA-C3 based on the dependency tree.
   Initiated by the initial FS compartment as "pull".
   No actions are required for Platform Management.
   The FS achieved the 2oo3 again and provides diagnosis data via interface I1-SDI to SFD.

As long as SCA is a “human person,” the recovery is not automated. For automation of SIL4 recovery a SCA-Tool is necessary.

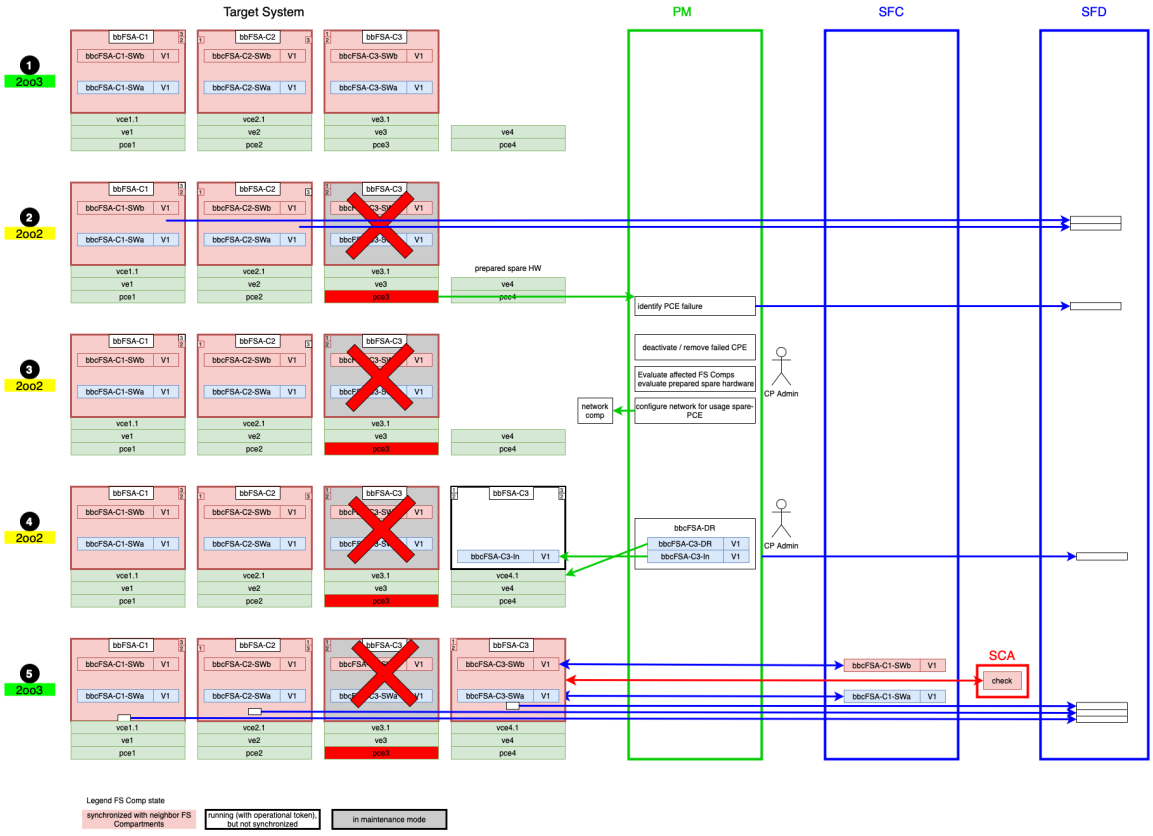


Figure 14 Recovery of a individual HW failure

[Open]

2.2.3.3.2 Recovery of a VE SW failure

SPT2CE-3580 - Recovery of an VE SW failure

The failure of a VE software is identified by the platform management.

The platform management is aware of the affected FS compartments installed on this failed CP VE software.

The recovery steps are the same as for the recovery of an individual CP hardware failure, see

SPT2CE-3089 - Recovery of Individual CP hardware failure.

[Open **]**

2.2.3.3.3 Recovery of VCE failure

SPT2CE-3568 - Recovery of an individual VCE failure

The failure of a VCE is identified by the platform management -

VCE management should know the status of VCE.

VCE management should provide a recovery mechanism for VCE, e.g., a restart.

VCE management should provide the VCE status to platform management.

The platform management is aware of the affected FS compartment installed on this failed VCE.

For the recovery of the affected FS compartment, the following steps are necessary:

1. Starting point: FS is running as 2oo3.
2. Failure of vce3.1 occurs and causes FS Comp-3 to stop.
   FS Comp-1/2 identify the missing neighbor FS Comp-3. This leads to reduced FS availability to 2oo2 mode.
   FS Comp-1 and FS-Comp-2 provide diagnosis data via interface I1-SDI to SFD.
   PM provides diagnostic data via the I1-SDI interface to SFD.
3. The platform management evaluates the required recovery mechanism: deploying the FS Comp-3 software into a new VCE on the same pce3.
   The failed vce3.1 with bbFSA-C3 is deleted to allow new creation of the bbID in a new vce3.2.
   Initiated by the CP Admin.
4. The Platform Mngmt uses the stored deploy rules bbcFSA-C3-DR and initial software bbcFSA-C3-IN and creates the initial FS Comp-3 as vce3.2 on pce3.
   PM provides diagnosis data for vce3.2 via interface I1-SDI to SFD.
5. All the functional software bbcFSA-C3-SWa and bbcFSA-C3-SWb are updated via I1-SMI into the FS Comp3 = bbFSA-C3 based on the dependency tree.
   Initiated by the initial FS compartment as "pull".
   No actions are required for Platform Management.

   As long as SCA is a “human person,” the recovery is not automated. For automation of SIL4 recovery a SCA-Tool is necessary.

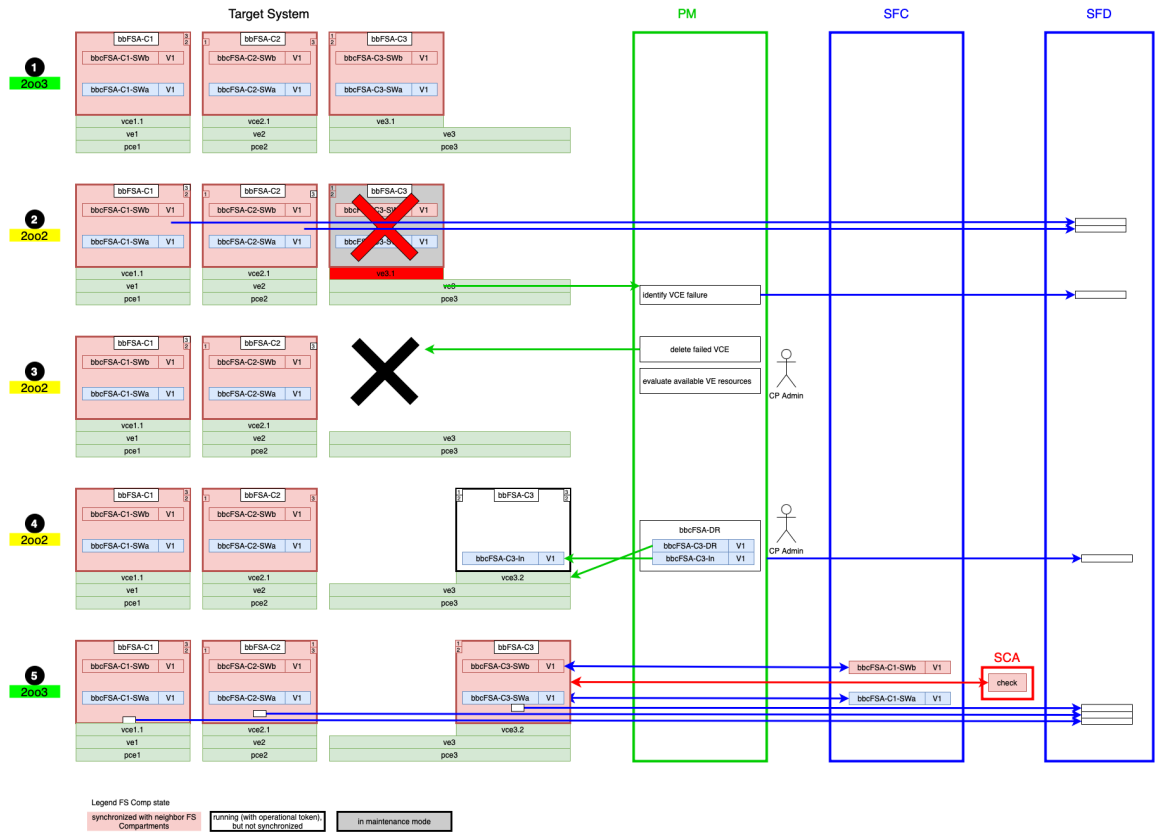


Figure 15 Recovery of a VCE failure

[Open]

2.2.3.4 Configuration failures by PM

2.2.3.4.1 Unintentional Duplication of an FS Comp

SPT2CE-3566 - Unintentional Duplication of an FS Compartment

When dealing with the basic integrity CP software environment and platform management, it can happen that an individual running the FS compartment is unintentionally duplicated.
A duplicated FS Compartment would run with **identical bbID** and **identical configuration** (using the same IP addresses).

From a safety perspective, it's not allowed for a bbID to exist multiple times in a critical responsibility. By this, the impact of an unintentional duplication of an FS compartment needs to be analyzed.

In the case of a SIL4 FS, the duplicated compartment would try to synchronize with the neighboring FS compartments.
Repeated use of identical IP addresses would result in communication failures.
The following cases are theoretically possible:

1. The duplicated compartment is not able to synchronize with neighboring compartment(s).
   In this case, the compartment does not achieve a "safe" running state and cannot establish safe communication via I0 with other systems.
   The duplicated compartment is running unsynchronized and isolated; it does not take over vital responsibilities.
2. The duplicated FS compartment successfully connects to the neighboring compartments.
   In this case, the safety layer would check if the duplicated FS compartment behaves in the same way as the original running compartments.
   Means the safety layer ensures that the running FS behaves safely, even in the presence of a duplicated compartment.

-> The unintentional duplication of a FS compartment does not lead to a unsafe FS state.

Depending on the safety layer's solution, it may identify an involved duplicate and deactivate both FS compartments (the duplicated new compartment and the originally running compartment). This would lead to reduced FS availability, e.g., 2oo2 mode (instead of 2oo3).
For this reason, to ensure availability, the duplication of running FS compartments should be prevented for SIL4 FS by additional intelligence within the platform management. **[** Open **]**

SPT2CE-3564 - The platform management checks the uniqueness of the bbID before creation of the new bbID as FS compartment with initial software.
If the bbID already exists the bbID is not created. **[** Open **]**

SPT2CE-3562 - The platform management monitors continuously the uniqueness of active running bbIDs.
If a duplicated bbID is identified both instances of this bbID are deactivated by the platform management. **[** Open **]**

SPT2CE-3574 - When a PCE starts up (e.g. after power on) the platform managements checks the uniqueness of the upstarting bbIDs.
It the upstarting bbID already exists (e.g. running on another PCE as recovery of a PCE failure) the upstarting bbID shall be stopped to avoid multiple existence of same bbID.
[Open **]**

2.2.3.4.2 Communication conflicts in case of FS compartment cloning

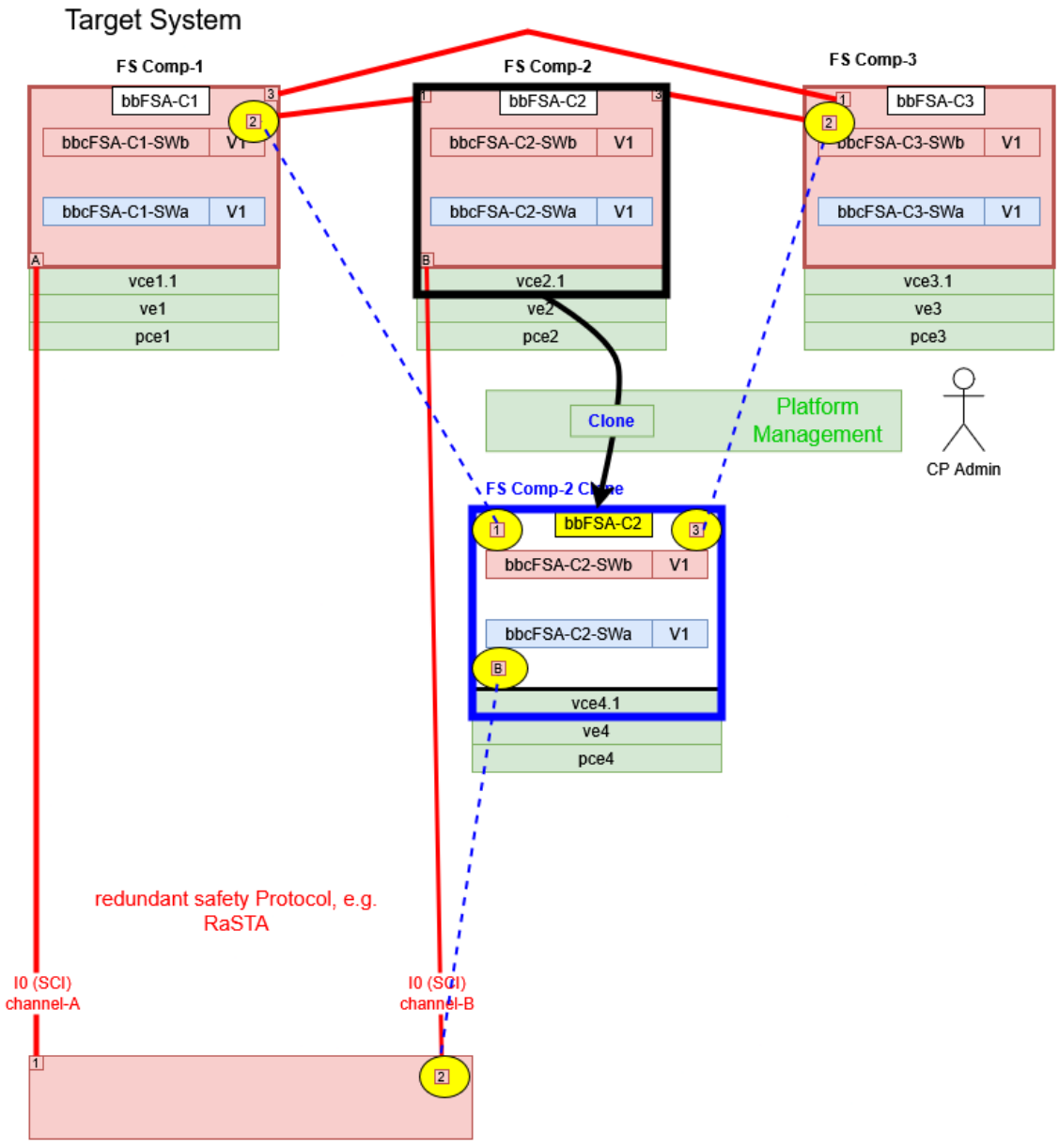
SPT2CE-3572 - Communication conflicts in the case of FS compartment cloning

Each cloned FS compartment runs with an identical configuration (e.g., identical IP addresses) as the original compartment.
This multiple use of the same IP addresses leads to communication failures during build-up.

Safety-related communication protocols, such as RaSTA, are realized as point-to-point communication and do not support additional endpoints. Safety protocol-specific safety measures, such as sequence counters within the communication channels, ensure that a cloned FS compartment cannot take over the safe communication channel without interruption and elicit a safe response from the communication partner. The cloned FS compartment would communicate with a lower (= older) sequence number than the original FS compartment,

leading to an interruption of the communication channel.

The figure below shows the communication architecture conflicts in the case of FS compartment cloning, exemplified for a SIL4 FS with a 2oo3 principle and redundant safe communication (I0) to an external system.



[Open]

3 Computing Platform Subsystems

SPT2CE-3526 - The Computing Platform Subsystems include CP Hardware and CP Software. **[** Open **]**

3.1 CP functions allocation to System/subsystems

SPT2CE-3527 - The chapter assigned the system functions described in the previous document 'System Analysis' to the CP sub systems.

CP Functions	Description	Subsystem allocation	Safety Related
CP as runtime environment for FS Comps			
SPT2CE-2484 - Fct-CP Basic - CP as basic integrity standard solution for SW and HW	Provide a Computing Platform as basic integrity solution based on COTS components to run FS compartments aggregated on the same standard hardware.	Process for CP	No
SPT2CE-1862 - Fct-CP Basic - Runtime environment for FS Comp	Provide runtime environment to run FS Compartments with flexibility in usage of guest OS within the FS Compartment.	CP HW CP SW	No
SPT2CE-2315 - Fct-CP Basic - Isolation of the resources mapped to different FS comps	Provide a resource isolation for the FS compartments. Each FS compartment shall use it's own mapped resources. The freedom from interference between the aggregated FS compartments shall be demonstrated through generic test environment.	CP HW CP SW	No
SPT2CE-2592 - Fct-CP Basic - Provide ID of the used physical CP HW node via I3-CP	Provide a unique HW identification of the currently used physical machine to the FS Compartment.	CP HW CP SW	Yes
SPT2CE-1901 - Fct-CP Basic - Provide steady clock from physical hardware of used CP HW node via I3-CP	Provide steady (monotonous) system clock from the physical machine to the FS Compartment.	CP HW CP SW	Yes
Configuration of the CP for usage by FS Comps			
SPT2CE-2505 - Fct-CP Config - FS Comp mapping to CP HW according to FS redundancy principle	Mapping of the FS Comps to CP hardware according to the FS redundancy principle as defined in the FS Comp deployment rules BBC(DR).	CP SW	No

CP Functions	Description	Subsystem allocation	Safety Related
SPT2CE-1911 - Fct-CP Config - FS related CP configuration according to deployment rules	Configure the Computing Platform according to the FS Comp related deployment rules BBC(DR) • needed runtime resources (CPU cores, memory, ..) • needed communication resources • connections to communication partners on separate physical machine (neighbour compartments of SIL4 FS) • connections to communication partners on any physical machine of the same own Computing Platform • connections to external communication partners (as e.g. decentralised object controllers)	CP SW	No
SPT2CE-2314 - Fct-CP Config - FS Comp related mapping of runtime resources	Provide FS Compartment related runtime resource mapping according to deployment rules which are provided by the FS Compartments. • needed cores • needed memory • needed internal I/O resources • needed storage	CP SW	No
SPT2CE-2327 - Fct-CP Config - FS Comp related mapping of communication resources	Provide FS Compartment related mapping of the communication resources (e.g. virtual ethernet ports, routing). The mapping shall be stable for running FS compartments. The installation of additional FS compartments may not have an impact to the resource mapping of the already installed FS compartments.	CP SW	No
SW handling for CP Software on CP Hardware			
SPT2CE-1899 - Fct-CP - Deploy CP Software onto CP Hardware	Create an instance of the Computing Platform Software onto a Computing Platform Hardware.	CP SW	No
SPT2CE-2503 - Fct-CP - Update CP software with compatibility for I3 and configuration	Support CP software update with compatibility at the I3 interface and configuration of existing FS compartments.	CP SW	No

CP Functions	Description	Subsystem allocation	Safety Related
SPT2CE-1984 - Fct-CP - Update CP software HW-wise	Support CP hardware-wise sequential update of the CP software with compatibility at the I3 interface to existing FS compartments.	CP SW	No
SPT2CE-1978 - Fct-CP - Backup CP Software and CP Configuration	Create a backup of the currently used CP software and CP configuration.	CP SW	No
SPT2CE-2328 - Fct-CP - Restore CP backup-version of the CP Software and CP Configuration	Restore a backup-version of the CP software and CP configuration.	CP SW	No
Handling of FS Comp SW to deploy and update SMI			
SPT2CE-2512 - Fct-CP SMI - CP with SMI service to provide interface I1-SMI for FS Comp deployment	The CP management provides the functionality (e.g. update service) to provide the interface I1-SMI for the deployment of a new FS Comp.	CP SW	No
SPT2CE-1850 - Fct-CP SMI - Creation of the Initial FS Comp with BBC(InSW)	Creation of a new FS Comp with BBC(InSW)) as initial software. Transfer of the BBC(DR) via I1-Update (SMI) from Shared Services to the Computing Platform.	CP SW	No
SPT2CE-2002 - Fct-CP SMI - Shutdown a FS Comp	Shutdown of a FS Comp.	CP SW	No
IT-Security SSI			
SPT2CE-2325 - Fct-CP SSI - Provide access to HW-based security solution e.g. TPM	FS compartment needs access to the HW-based security solutions e.g. TPM of the used PCE.	CP HW CP SW	No
SPT2CE-2449 - Fct-CP SSI - Restriction and protection of the device-internal communication	The ability to restrict and secure the communication of an FS Compartment to other compartments the same FS. Only compartments with a security stack are allowed to participate in communications with other functional systems even on a same hardware.	CP SW	No
SPT2CE-2562 - Fct-CP SSI - Provide I3 with interface for secure communication of the FS Comp	Provide I3 with interface for secure communication of the FS Comp.	CP SW	No

CP Functions	Description	Subsystem allocation	Safety Related
SPT2CE-2563 - Fct-CP SSI - Provide I1-SSI by CP	The CP provides interfaces to at least the following IT-security services: • Time server • Logging • Certificate-handling	CP SW	No
Diagnostics SDI			
SPT2CE-2431 - Fct-CP SDI - State monitoring of the CP hardware nodes	Monitoring of the states of the individual CP hardware nodes. Note: State is provided as diagnostic data via interface I1-Diagnostics to the Shared Service Diagnostics.	CP HW CP SW	No
SPT2CE-2433 - Fct-CP SDI - State monitoring of the network communication	Monitoring of the states of the network paths and interfaces within the CP (communication between FS Comps running on CP). Note: State is provided as diagnostic data via interface I1-Diagnostics to the Shared Service Diagnostics.	CP HW CP SW	No
SPT2CE-2323 - Fct-CP SDI - Process root cause analysis for FS Comp failures	The CP shall provide data needed to support a root cause analysis in the CP or FS. Relevant diagnostic data includes: • state of the individual FS compartment • state of the CP software (relevant for the FS compartment) • state of the CP hardware (which is relevant for the FS compartment) • state of the communication network (which is relevant for the FS compartment)	CP HW CP SW	No

CP Functions	Description	Subsystem allocation	Safety Related
SPT2CE-2322 - Fct-CP SDI - Provide CP diagnostic data according I1-Diag SDI	Computing platform provides its own health state information to the shared services through I1: • states of the individual CP Software instances (running on CP hardware nodes) • states of the individual physical CP hardware nodes • states of the network components (which are needed for communication between FS compartments)	CP HW CP SW	No
Recovery			
SPT2CE-1848 - Fct-CP Rec - Automatic Restart of a existing FS Comp which has failed	Automatic Restart of existing FS Compartment.	CP SW	No
SPT2CE-2527 - Fct-CP Rec - Automatic deployment of a failed FS Comp on another CP location	Automatic Deployment of a failed FS Comp on another CP location.	CP SW	No
SPT2CE-2398 - Fct-CP Rec - Repair of a SW failure within the CP software	Repair of a SW failure within the CP Software e.g., by restarting the CP SW on one PCE.	CP SW	No
SPT2CE-2561 - Fct-CP Rec - Deletion of FS Comp	Deletion of a FS Comp.	CP SW	No
SPT2CE-2402 - Fct-CP Rec - Deactivate an individual CP hardware node	Deactivation of an individual physical CP hardware node. Open safety aspect: relevant to avoid "split-brain" in context of geographical redundant CPs	CP HW	No
SPT2CE-2003 - Fct-CP Rec - Shutdown the entire CP	Shutdown of the total CP, means stop all CP software running on any CP hardware. Open safety aspect: avoid split-brain in case of switch-over of the operation to another CP placed on another location	CP HW	No

[Open]

3.2 Computing Platform Hardware (CP HW)

3.2.1 CP HW Subsystem Description

SPT2CE-3357 - Computing Platform Hardware consists of all the HW resources associated with the PCEs' HW. Additionally, all the network equipment associated with the CP internal architecture is part of the CP HW.
CP HW provides the following resources:
- Processors with hardware assisted virtualisation
- Memory with error correction
- Storage space
- LAN/Network/Switches/Routers

CP HW resources are managed with the general purpose COTS Platform Management SW.

CP HW provides the following types of physical interfaces:
- network interface
- power supply interface
- local diagnostic and maintenance interface (optional)

Note: This is general purpose COTS hardware and therefore no safety requirments/functions are applicable/required. **[** Open **]**

3.2.2 CP HW Failures Description

SPT2CE-3528 - The following failure modes are considered for CP HW:

Generic Failure Mode	CP HW failure description	Impact at CP system level / PM	Impact at FS level	Impact at SS level

Generic Failure Mode	CP HW failure description	Impact at CP system level / PM	Impact at FS level	Impact at SS level
No Function /data/message	Case 1: PCE HW not operational (cores, memory, storage, network failure, power supply) affecting the whole, or parts of PCE (indirectly, all VCEs and FS-Comps are not operational). Case 2: Network HW not operational (switches, routers) affecting the whole or parts of the PCE (indirectly, all VCEs and FS-Comps are not operational). NOTE: The HW failure can affect one, multiple, or all VCEs on a PCE; the failure identification and recovery is identical.	PM identifies that part of, or the entire PCE, is not operational/the network is not available. Affected VCEs and corresponding FS-Comps are identified, status is available at SS via PM SFD. PM starts creating new VCEs and configuring new FS-Comp based on BBC(DR) and BBC(InSW).	Case 1: FSs running on FS-Comps affected by a singular PCE HW failure are still operational with reduced redundancy, FS SFD information on reduced redundancy is available. Case 2: FSs running on FS-Comps affected by PCE HW failures are not operational, and no FS SFD information is available.	PCE HW failure status is available via PM SFD. When new FS-Comp are available, SS starts the SFC 'Deploy FS'.
Deletion of function/data / message	For CP HW failures, 'Deletion of function/data' failure is equivalent to 'No function/data'. Deletion of single or multiple messages due to VCE or network failures will be identified at the FS level.	See: 'No Function/data/message.' Additionally, temporary failures that affect only 1 message will not be identified by the PM.	See: 'No Function/data/message.' Additionally, temporary failures do not affect the FS level due to the FS availability implementation. Multiple/consecutive message deletions will be identified as a network failure.	See: 'No Function/data/message.'

Generic Failure Mode	CP HW failure description	Impact at CP system level / PM	Impact at FS level	Impact at SS level
Untimely (i.e., wrong moment, too soon, too late, too quick, too long) function/data / message	Untimely events at the CP HW level are due to failures of the system clock (affecting all VCEs) and timers (affecting a single VCE).	PM **can** identify the untimely events by implementing timer monitoring (optional).	FSs **shall** identify untimely events caused by the system clock via the FS safety layer. FS(i) will go into a safe state, and the FS FSD is provided to SS and I3.	SS shall request 'Deploy FS' on a new PCE for all VCEs/FS-Comp affected.
Corruption function / data / message	Corruption functions **cannot** be identified at the CP HW or VCE level (CPU/cores internal functions failures, cache and memory integrity failures if not covered by ECC, storage integrity failure). CP HW can comply with a minimum set of quality requirements.	Corruption functions **cannot** be identified by PM.	FSs shall identify the corruption functions that are implemented with the safety layer.	FSs provide the status to SS related to corruption functions and the FS operational status. SS will remove the FS(i) affected by the failure and will request a new VCE and FS-Comp environment. When new FS-Comp are available, SS starts the SFC 'Deploy FS'.
Freeze of function / data / message	For CP HW failures, 'freeze of function/data/message' failure is equivalent to 'No function/data/message' failure. Ex: intermittent freeze of a core under high load, failed cache lines, due to thermal degradation or quality issues	See: 'No Function / data / message'	See: 'No Function / data / message'	See: 'No Function / data / message'

Generic Failure Mode	CP HW failure description	Impact at CP system level / PM	Impact at FS level	Impact at SS level
Repetition of function / data / message	Repetition of functions/data/messages **cannot** be identified at CP HW or VCE level.	Repetition of functions/data/messages **cannot** be identifiedby PM.	FSs shall identify the repetition functions with the safety layer implemented. FS shall manage repetition functions internally.	FSs provide the status to SS via SFD.
Insertion of data / message	No impact/no source at CP HW level		FSs shall identify the insertion of data/messages within the safety layer implemented.	FSs provide the status to SS via SFD.
Re-sequencing of data/message	No impact/no source at CP HW level		FSs shall identify the resequencing of data/messages with the safety layer implemented.	FSs provide the status to SS via SFD.
Masquerade (i.e., security threat) of function/data / message	No impact/no source at CP HW level		FSs shall identify the masquerade with the security layer implemented.	FSs provide the status to SS via SFD.
Part of function / data / message	Detection of partial functionality is **limited** at the CP HW level. Ex: intermittent crash of a core under high load, memory single bit errors corrected by ECC, or ECC induced errors, intermittent read/write failures (memory, storage), network interfaces partially operational, data corruption under high speed transfer.	Detection of partial functionality is **limited** a t the PM level.	FSs shall identify the partial functions (sporadic miscomputations, wrong outputs) with the safety layer implemented.	FSs provide the status to SS via SFD.

[Open **]**

3.2.3 CP HW PRAMS Requirement

SPT2CE-3364 - The PRAMS requirements applicable to CP HW will be included in the later version of the document. **[** Open **]**

3.2.4 CP HW Cybersecurity

SPT2CE-3498 - Cybersecurity requirements applicable for CP HW to be introduced after completion of the agreed scope with the Security Domain and definition of the secure component.
The allocation of cybersecurity requirements (as defined in the EU Rail cybersecurity requirements v1.0 and subsequent versions 1.1 and 1.2 - OT security based on IEC62443), will require a cybersecurity architecture definition including FS and CP, based on the following principles:
- Standard cybersecurity requirements and cybersecurity architecture of a generic data center (public or private). Data center cybersecurity is based on IT security and complies with IEC 27001.
- Definition of secure components from the perspective of EU Rail cybersecurity requirements v1.0.
- Separation of safety and cybersecurity functions
- Management of cybersecurity-related updates outside of the safety-related SW (FS) **[** Open **]**

3.2.5 CP HW Diagnostic Requirements

SPT2CE-3367 - The CP HW shall provide the HW related diagnostic data defined in the CP Data model.
Note: A data model for onboard use-cases will be developed together with the TrainCS group.
[Open **]**

3.2.5.1 Diagnostic CP HW

SPT2CE-3369 - The allocation of the diagnostic functionality to monitor the CP hardware has to be evaluated.
SPT2CE-2431 - Fct-CP SDI - State monitoring of the CP hardware nodes **[** Open **]**

3.2.5.2 Diagnostic Network Interfaces

SPT2CE-3371 - The allocation of the diagnostic functionality to monitor the network interfaces has to be evaluated regarding the dependency on the interfaces I1 and I3.
SPT2CE-2433 - Fct-CP SDI - State monitoring of the network communication **[** Open **]**

3.2.6 CP HW Configuration Management

SPT2CE-3373 - CP HW configuration management is realised with PM and the virtualisation environment, based on FS-DR.
CP HW cannot be configured directly through the SMI. **[** Open **]**

3.3 Computing Platform Software (CP SW)

SPT2CE-3377 - The CP software has three main components, the Virtualisation Environment, VCE management, and Platform management.

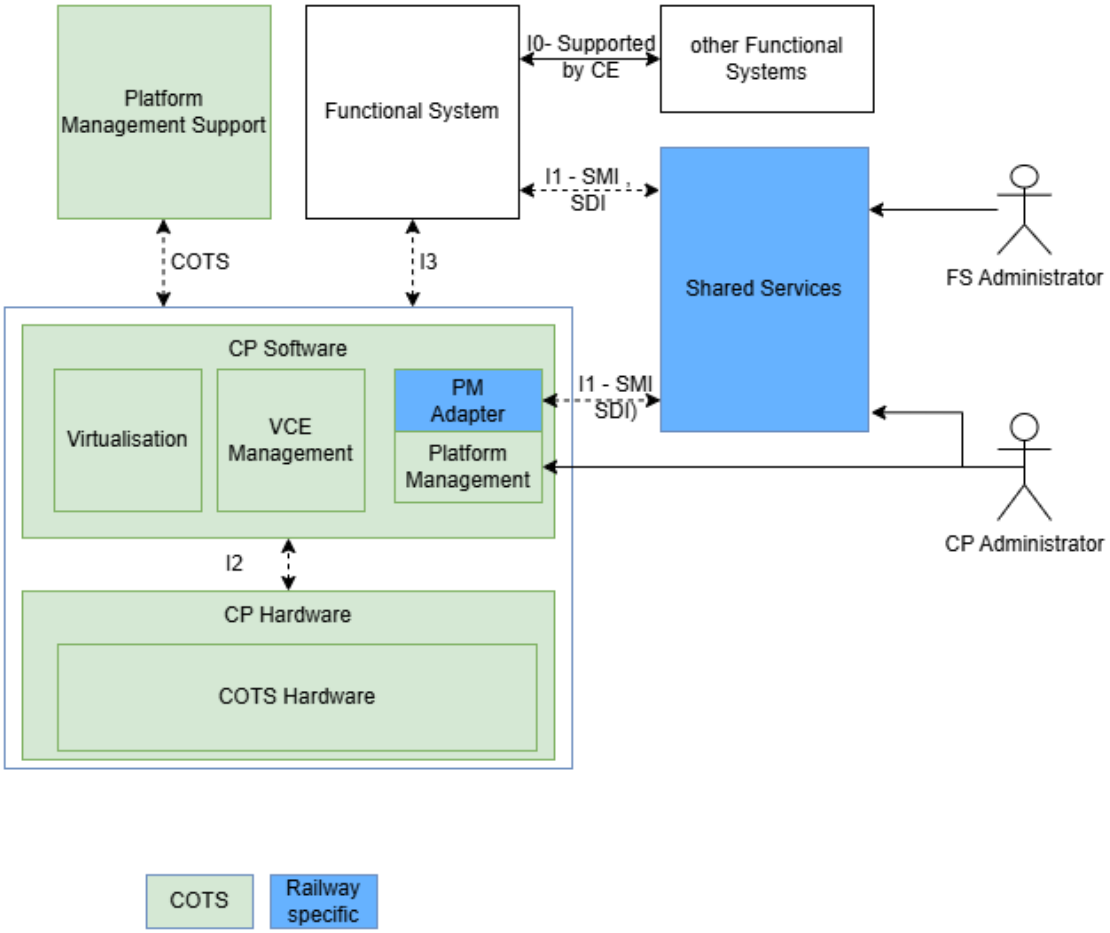


Figure 16 Platform Management

[Open **]**

3.3.1 Virtualisation environment

SPT2CE-3376 - Virtualisation provides the abstraction of physical computing resources to the FS compartment. It provides functions to start and stop compartments that meet the isolation needs and constraints provided by the VCE management.
Virtualisation comprises of multiple functions, which, as a whole, work together to provide this functionality:

- **Resource isolation:** Isolating, if available, the hardware resources requested during compartment creation based on the FSDR.
- **Hardware abstraction:** Provide a virtual representation of hardware to the compartment based on the FSDR needs.
- **Compartment lifecycle management:** Allows starting/stopping of compartments.

Note: Resource isolation is provided by a Hypervisor. This can run either directly on the hardware (Type 1) or as an application on the operating system (Type 2). There are some solutions that blur this line, which are integrated into a general-purpose operating system's kernel. For simplicity, we consider these to be of Type 1 as well. For the determinisim it is expected that Type-1 Hypervisor would be required.
Note: For resource abstraction, we consider only hardware-assisted options. While we are aware that full simulation is technically possible, we do not consider this feasible for safe systems at this time.
Note: Although containerisation is widely used, this solution was not chosen for reasons of strong isolation. **[** Open **]**

SPT2CE-3375 - There are several virtualisation solutions available on the market. As a major objective of the modular computing platform is to use COTS technology as much as possible, the Computing Platform targets the use of one or more COTS solutions, depending on the Functional System requirements.
Usually, a general purpose COTS virtualisation environment is a combination of several components:

- A hypervisor software for running the Virtual Computing Elements (the hypervisor needs to run the NHA natively to support safe compartments)
- A Virtual Computing Element management software to manage the lifecycle of Virtual Computing Elements on a Physical Computing Element
- A Platform Management Software to manage a fleet of Physical Computing Elements with (possibly different) hypervisors and VCE management as one.

The functionalities of this software can vary depending on the concrete implementation, and these elements can be provided by a single vendor or different vendors, depending on operator needs, for example Platform Management could be provided by a third party vendor specialised in datacenter management software which interfaces with the Virtual Computing Management Software through an interface specific to this solution or a more generic abstraction. **[** Open **]**

3.3.2 VCE management

SPT2CE-3380 - Virtual computing element manages the compartment. VCE management offers functions to create, configure, and remove compartments. VCE management has its own processes, tools and policies that shape its functions and enable the control and operation of the compartments.

The core functions are as follows:

- **Provision and Deployment**: Creating and deploying the new compartments based on the FSDR. The deployment can be either manual (one at a time) or automated (multiple compartments at once).

- **Configuration management**: Check and validate the compartment configuration (AEE and RTE settings, security policies).

- **Diagnostic and Performance management**: Logging and tracking health during runtime (CPU, memory, network utilisation).

- **Maintenance**: Apply security patches and software updates - and back up for recovery.
- **Decommissioning**: Shutting down and removing compartments.
- **Resource management and orchestratio**n: Resource allocation, setting, and managing physical resource guarantees.
- **Inventory**: Maintaining the real-time list of existing compartments and their configuration.

[Open **]**

SPT2CE-3382 - The deployment mechanism for FSDR is defined in VCE management, and CP should ensure that there are sufficient resources available in order to create the compartment(s) on each CE.
SPT2CE-1862 - Fct-CP Basic - Runtime environment for FS Comp **[** Open **]**

3.3.3 Platform Management

SPT2CE-3384 - The platform management is part of the computing platform software and serves to manage the Compartment Execution Environment for the FS Compartments.
It consists of:

- The general purpose COTS virtualisation management functions, and
- Railway specific functions necessary to implement the interface to Shared Services.

The configuration management fundamentally distinguishes between:

1. **FS Compartment Deployment**
   This refers to the setup of the FS Compartments according to the deployment rules provided by the FS Compartments, with an initial FS Compartment software.
   This deployment is performed by the Shared Services via the I1-SMI interface to the platform management of the computing platform.
2. **FS Compartment Update**
   This refers to loading the functional FS Compartment software into the initial FS Compartment (as deployed, see above).
   This update is performed by the Shared Services Update via Interface I1 – SMI into the initial FS Compartment, without dependency on the platform management.

[Open **]**

SPT2CE-3385 - Conversion of diagnostic data from solution-specific to SDI railway-specific, can be implemented inside or outside the computing platform software.
Two scenarios:

- One is COTS generic diagnostic data not relevant for CP,
- other scenario data relevant for update and recovery (state of the compartment
  - Data model between PM and SS
  - equipment model for the FS compartment

[Open **]**

SPT2CE-3499 - SPT2CE-2503 - Fct-CP - Update CP software with compatibility for I3 and configuration
SPT2CE-1984 - Fct-CP - Update CP software HW-wise
SPT2CE-1978 - Fct-CP - Backup CP Software and CP Configuration
SPT2CE-2328 - Fct-CP - Restore CP backup-version of the CP Software and CP Configuration
SPT2CE-2512 - Fct-CP SMI - CP with SMI service to provide interface I1-SMI for FS Comp deployment

Functions for recovery by Platformchapter 3.1 Management:
SPT2CE-1848 - Fct-CP Rec - Automatic Restart of a existing FS Comp which has failed
SPT2CE-2398 - Fct-CP Rec - Repair of a SW failure within the CP software
SPT2CE-2561 - Fct-CP Rec - Deletion of FS Comp
SPT2CE-2402 - Fct-CP Rec - Deactivate an individual CP hardware node
SPT2CE-2003 - Fct-CP Rec - Shutdown the entire CP
SPT2CE-2512 - Fct-CP SMI - CP with SMI service to provide interface I1-SMI for FS Comp deployment
SPT2CE-1850 - Fct-CP SMI - Creation of the Initial FS Comp with BBC(InSW)
SPT2CE-2002 - Fct-CP SMI - Shutdown a FS Comp **[** Open **]**

3.3.4 CP Software Failures Description

SPT2CE-3530 -

Understanding of the CE SW failure modes are crucial for system high availability and efficient recovery. The following failure modes are considered for CP SW:

Failure Type	Impact on CP SW	CP SW failure description	Impact at the CP system level / PM	Impact at the FS level	Impact at the SFD level

Failure Type	Impact on CP SW	CP SW failure description	Impact at the CP system level / PM	Impact at the FS level	Impact at the SFD level
Hard	Complete CP SW failed	The CP SW is not operational (error in virtualisation, VCE management, PM). All VCEs, FS Comp, and PM are affected.	This results in disaster scenarios in which the entire data center, including the entire platform software, is down. To handle such cases fallback mechanism should be implemented. Here solution like geographical redundancy would be required.	All FSs are unavailable /not operational due to the complete CP SW failure.	SFD should indicate failure due to not receiving data/being unable to connect.
Soft	Partial CP SW is not available	The CP SW is partially not operational (error in virtualisation, VCE management).	PM notifies SFD of the details of the affected VCEs.	PM and FS notify SFD of the details of the affected FSs.	SFD received the details of affected VCEs and FSs through SDI.

[Open **]**

SPT2CE-3529 -

Failure modes at CE SW components are categorised as follows:

Component	Possible Causes	Impact
Hypervisor	• Hypervisor Crash • Bugs • Patch failure • Scheduler Failure • Management agent failure • Memory Over commit	Multiple VCEs can go down simultaneously
Network	• Virtual switch misconfiguration • VLAN misconfiguration • DNS failure	VCEs not reachable, high latency

Component	Possible Causes	Impact
Storage	• Storage medium down • I/O latency • storage medium is full •	VCEs can freeze, down or degraded performance
Orchestration	• VMs deployment/re-deployment fails • API failure • wrong image deployment	VCEs are not available
Security	• Credential compromised • Malware infection • DDoS attack	VCEs down and FS not available

[Open **]**

SPT2CE-3500 - For all CP SW failures it's in the responsibility of the FS to identify safety relevant CP SW failures and react according to its own safety requirements. The details in the context of failure detection and reaction depend on the safety concept of the solution specific safety layer. A FS reaction may lead to an individual FS compartment stop/restart or even to a total FS stop/restart. **[** Open **]**

4 Interfaces

4.1 I2-Hardware Compatibility Interface

SPT2CE-3501 - I2 is a hardware compatibility interface between CPHW and CPSW. The detailed technical specification of I2 interface is provided in a separate I2-Hardware Abstraction Interface Specification document. **[** Open **]**

4.2 I3-Virtualisation Interface

SPT2CE-3531 - I3 is a interface between the FS compartment and the virtual computing element. It has two parts: the hardware abstraction and the minimum data exchange needed between FS compartment and VCEs to operate FS.
I3 Hardware abstraction should be general purpose COTS solution. This chapter describes the additional functions required by the functional system to operate on the virtual computing element. **[** Open **]**

4.2.1 Mapping of runtime resources

SPT2CE-3505 - CP provides the mapping of runtime resources for each FS compartment. Interface I3 provides access of FS to the allocated resources of a compartment.

Mapping of runtime resources

The FS provides the FSDR considering required runtime resources and the redundancy principle of the FS.

The FSDR shall be considered in the configuration of the CP.

SPT2CE-2505 - Fct-CP Config - FS Comp mapping to CP HW according to FS redundancy principle

SPT2CE-2314 - Fct-CP Config - FS Comp related mapping of runtime resources

SPT2CE-1911 - Fct-CP Config - FS related CP configuration according to deployment rules **[** Open **]**

SPT2CE-3504 -

The CP shall support the isolation of mapped resources to provide a stable runtime environment to each FS compartment.

Resources should not be shared between compartments to achieve highest FS availability.

Note: All resources assigned to VCEs should be exclusively available.

SPT2CE-2315 - Fct-CP Basic - Isolation of the resources mapped to different FS comps **[** Open **]**

SPT2CE-3507 - The NHA within the CP software on each PCE shall provide a unique HW identification of the used CP hardware to all FS Comp running on this hardware.

SPT2CE-2368 - REQ-CP - NHA - Provide unique identification of the used CP hardware to the FS Comps **[** Open **]**

SPT2CE-3506 -

All compartments created on a CP should have a unique Building Block Identifier.

Each PCE of CP should have a unique identifier. **[** Open **]**

SPT2CE-3503 -

Communication resources required for each compartment, are defined in the FSDR.

CP shall ensure that there is sufficient bandwidth available for communication resource(s) in order to be assigned to the compartment(s) on each CE.

SPT2CE-2327 - Fct-CP Config - FS Comp related mapping of communication resources **[** Open **]**

4.2.2 CPU Cores/Cache Memory

SPT2CE-3434 - Implementation of dedicated cores for each compartment (core pinning) is recommended, but it is not always possible.

Use of cores dedicated cache memory is implicit.

Use of cores shared cache memory should be evaluated from performance and safety point of view, and can be part of FS deployment rules.

CPU resources required for each compartment, including performance requirements, are defined in the FSDR. **[** Open **]**

4.2.3 Memory/Memory channels

SPT2CE-3416 - Memory resources required for each compartment, including performance requirements, are defined in the FSDR. **[** Open **]**

4.2.4 Storage

SPT2CE-3415 - Storage resources required for each compartment, including performance requirements, are defined in the FSDR. **[** Open **]**

4.2.5 Watchdog

SPT2CE-3414 - A compartment watchdog shall be implemented if required by the FS. Each FS instance shall trigger its compartment instance watchdog. The watchdog monitors the correct operation of FS.
In case that the watchdog is not triggered, a mechanism to stop the FS shall be implemented.
The watchdog shall be disabled during the boot time of the FS.
Watchdog parameters are defined in the FSDR. **[** Open **]**

4.2.6 NHA

SPT2CE-3440 - A compartment Native Hardware Access (NHA) shall be implemented. Each FS instance shall monitor its compartment instance NHA status.

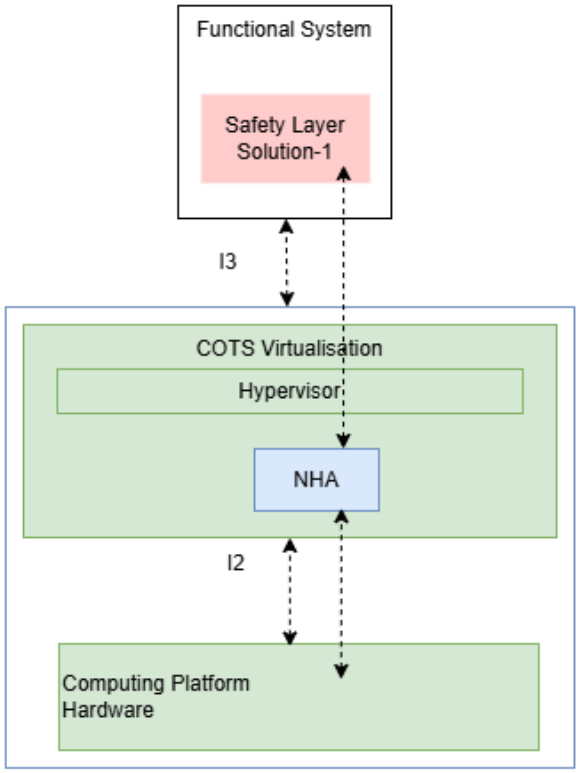
Presumed information provided by NHA:

- Unique identification of the physical hardware devices SPT2CE-2592 - Fct-CP Basic - Provide ID of the used physical CP HW node via I3-CP
- Steady clock input source from the physical hardware with controlled drift SPT2CE-1901 - Fct-CP Basic - Provide steady clock from physical hardware of used CP HW node via I3-CP
- Further information depending on the concrete safety concept of a solution of a safety layer

NHA parameters are defined in FSDR.
NHA should use the data available from CPHW NHA.
CPHW NHA parameters are defined in interface I2.

[Open **]**

SPT2CE-3439 - FS needs to realise time related safety critical services as e.g. cyclic and synchronous replica processing or timer-services. The time related safety functions are implemented using the compartment instance system clock, which is provided by NHA.
The FS safety layer needs at least 2 independent monotonous clock input sources. **[** Open **]**



SPT2CE-3438 -

Figure 17 Native Hardware Access (NHA)

[Open **]**

4.2.7 System Clock

SPT2CE-3437 - System clock requirements are defined in chapter 4.2.6 NHA. **[** Open **]**

4.2.8 System Time and Date

SPT2CE-3491 - PM provides the system time and date through NTP via SSI interface for diagnostic related data. **[** Open **]**

4.2.9 CP Boot/Secure Boot and RoT

SPT2CE-3443 - Each compartment instance shall implement a secure boot (SB) mechanism.
Comp instance SB shall be included in the overall SB and root of trust (RoT) mechanism, of the CE.
FS compartments should be signed. **[** Open **]**

4.2.10 Communication Networks

SPT2CE-3442 - CP communication networks are implemented via Ethernet.
Each compartment implements the communication network (VLANs), based on FSDR requirements. CP allocates the physical ports (LANs) based on FSDR security (and possibly performance) requirements.

QoS marking may be needed to manage the quality of service.
If DSCP traffic marking is used an additional categorie "FS internal communication" may be needed. **[** Open **]**

SPT2CE-3441 - Communication Types and Message Priorities

The table below shows a non-exhaustive list of different communication types with belonging message categories.

Communication Type	Message categories
PRIV = private proprietary communication between FS compartments of the same FS.	• operative messages between FS compartments • recovery synchronisation between FS compartments (e.g. after restart of individual FS compartment)
SCI = standardised operative communication between Functional Systems	• communication control messages (e.g. heart-beats) • operative data messages
ETCS = standardised operative communication between Function Systems and trains	• communication control messages (e.g. heart-beats) • operative data messages
SMI = standardised communication to Shared Service Configuration **NOTE:** Possibly additionally COTS protocols are enabled by the SMI	• messages to deploy initial SW via PM • messages to update SW into initial FS Comp
SDI = standardised communication to Shared Service Configuration **NOTE:** Possibly additionally COTS protocols are enabled by the SDI	• diagnosis commands • diagnosis messages
PM = COTS specific comm. between VE and Platform Management	• diagnosis messages from VE to PM • deploy messages from PM to VE

[Open]

SPT2CE-3436 - Communication partners

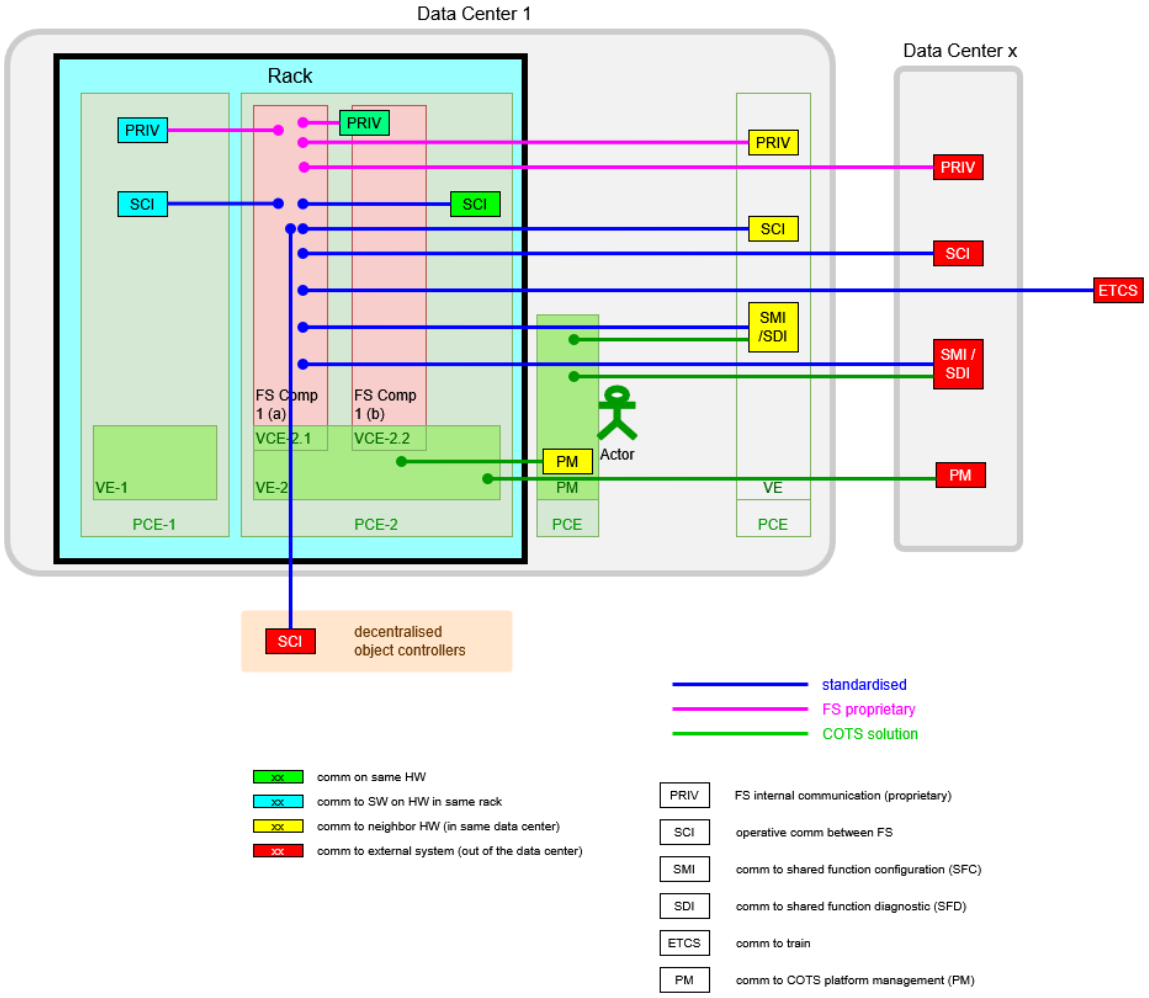


Figure 18 Communication types and partners

Example Figure 8 :

2 Functional SIL4 Systems FSA and FSB within the same Data Center 2.

Each FS running as 2oo2 (3rd channel for redundancy is not shown in the figure to keep the figure simple).

Each FS with PRIV communication on same HW and private communication to neighbour HW.

Both FS communicating with each other via SCI.

Both FS communicating with external systems (object controllers, traffic management) via SCI.

PM placed in same Data Center 2 on own PCE.

Shared Services placed in another Data Center 1.

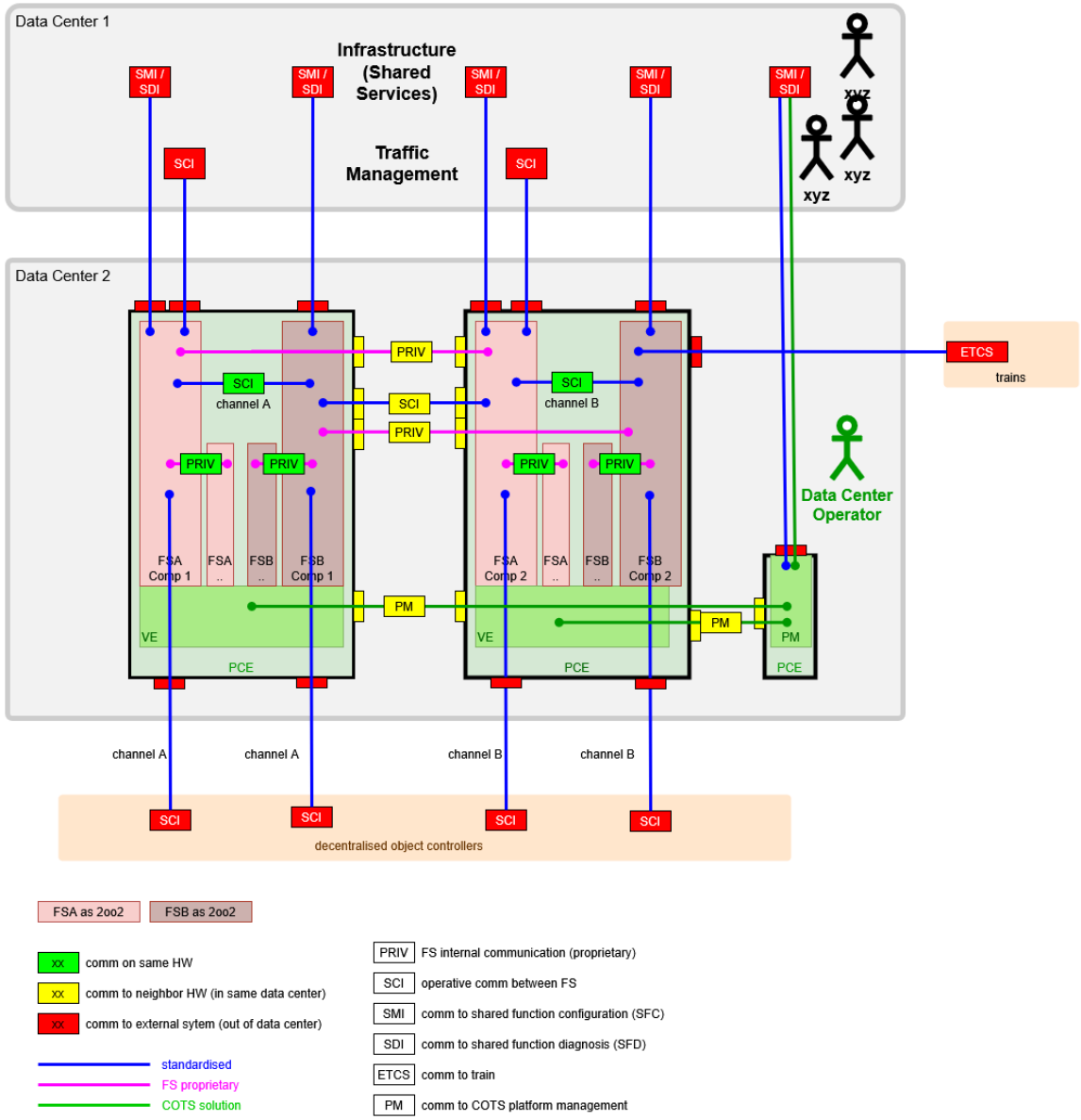


Figure 19 Exemplary Configuration Data Center

BB	category	type	comm partner	OWN HW	OTH HW	OTH RACK	EXT
FS Comp-x	PRIV	operational	Other Comp of same FS	X	X	X	X
FS Comp-x	PRIV	recovery	Other Comp of same FS	X	X	X	X
FS Comp-x	SCI	operational	other FS	X	X	X	X
FS Comp-x	ETCS	operational	train	./.	./.	./.	X

BB	category	type	comm partner	OWN HW	OTH HW	OTH RACK	EXT
FS Comp-x	SSI	logging	SSI	./.	X	X	X
FS Comp-x	SMI	config	SFC	./.	X	X	X
VE-x	PM	config	COTS PM-x	./.	X	X	X
VE-x	PM	diag	COTS PM-x	./.	X	X	X
COTS Diag-x	PM	diag	COTS PM-x	X	X	X	X
COTS PM-x	PM	diag	SFD	./.	X	X	X
COTS PM-x	SMI	config	SFC	./.	X	X	X
COTS PM-x	SSI	time	SSI	./.	X	X	X
VE-x	PM	time	COTS PM-x	X	X	X	X

In context of SW maintenance a flexibility is necessary in context of re-location of software on PCEs.
This means that the relationship "comm. partner is on own HW / other HW / other Rack / ext" may be changed during the lifetime of the systems - **each communication relationship may be or become an "external" communication** (even PRIV communication may be external in context of geographical redundancy).
This flexiblity in communication relationsships shall be supported by the IT security architecture.

For an efficient handling of all IT security related communication mechanism the secure communication shall be provided by the Computing platform as system-/vendor-independent service.
[Open **]**

SPT2CE-3433 - REQ-HLPI-16 - The VE shall support the mapping of VCEs to virtualised Ethernet adapters and the alignment of virtualised Ethernet adapters to physical Ethernet cards of the PCE. **[** Open **]**

4.2.10.1 Other Communication Interfaces

SPT2CE-3432 - Any physical interface other than Ethernet/LAN available at CP instance is disabled by default and cannot be used by FS.
This is ensured by the hardening requirements of the CP. **[** Open **]**

4.2.11 File System Virtualisation

SPT2CE-3422 - The general purpose COTS Virtualisation shall provide an abstraction for block devices that maps FS Compartments onto files or other means of representing a virtual block device. I3 shall ensure that this device abstraction behaves like a regular block device to the FS Compartments and does not impose any restrictions on filesystems used in the FS Compartment. **[** Open **]**

SPT2CE-3420 - SPT2CE-2314 - Fct-CP Config - FS Comp related mapping of runtime resources
SPT2CE-2178 - Sys-2-FSComp - Provide runtime environment to FS Comp **[** Open **]**

5 Open Items

SPT2CE-3460 - Architecture aspects related to configuration management, diagnostic and security, FSDR will be covered in the later version of system architecture document with I1-interface specifications. **[** Open **]**

SPT2CE-3464 - The Security architecture and security requirements will be covered in the later version of system architecture document with I1-interface specifications. This might also impact I3 interface specially communication between two FS compartments running on same hardware. **[** Open **]**

SPT2CE-3463 - The Diagnostic data model for the onboard will be develop by the TrainCS group as their product group model. **[** Open **]**

SPT2CE-3508 - PRAMS requirements will be added in a future version of this document after the EET process for PRAMS requirements allocation will be published. **[** Open **]**

SPT2CE-3510 - The safety process to integrate FS with NHA needs to be formulated. **[** Open **]**

6 Conclusion

SPT2CE-3509 - This document provides the CE System Architecture, which describes a strategic shift toward a standardised, high-integrity computing environment for the CCS system. The core objective is to move away from monolithic architecture and instead adopt a modular framework that separates the Functional System from the underlying physical infrastructure.

Key Takeaways:

1. **The Separation of Hardware and Software**: The most significant takeaway is the separation of the Software stack from the hardware through standardised interfaces. By using virtual computing elements, the system not only allows sharing the hardware among FSs but also allows railway applications (Functional Systems) to run on generic Commercial Off-the-Shelf (COTS) hardware. This decoupling means that as hardware becomes obsolete, the software can be migrated to newer servers without requiring a complete redesign of the safety logic.
2. **A Focus on SIL4 Safety Standards**: Despite using standard commercial hardware, the architecture is designed to meet Safety Integrity Level 4 (SIL4-highest level of reliability in railway operations).
Safety is ensured by Safety Layer, together with the Native Hardware Access component. The safety layer bridges the gap between fail-safe requirements on the standard COTS hardware.
3. **Standardised Interfaces (I2 and I3)**: To enable the use of COTS hardware and software, the architecture emphasizes the importance of two interfaces, namely:

Interface 12 (I2): Provides the hardware compatibility list, including the hardware functions required to run general purpose COTS virtualisation software and how the software interacts with physical hardware (CPUs, memory, network cards).
Interface 13 (I3): provides a standardised interface between the Computing Platform (CP) and the Functional System (FS) and enables the Functional System to be independent of a specific implementation of the computing platform hardware.
4. **Resilience and Failure Management**: The CE system architecture provides mechanisms for automatic deployment and recovery that provides resilient to hardware and software failures. It includes diagnostic frameworks (Interface I3-Diag and I1-Diag) that monitor the system's health at runtime. If a physical server fails, the platform automatically redeploys FSs to spare hardware, maintaining operational continuity even in the event of a technical failure (will be descrined in the later version of the document).

In a nutshell, the proposed architecture serves as a blueprint for a flexible, cost-effective, and interoperable European system. By moving toward a virtualised, hardware-agnostic system, the rail industry can adopt the rapid innovations of the IT world while maintaining the uncompromising safety standards required for public transportation. The shift toward a Modular Platform ensures that the rail system remains maintainable and scalable in the future.
[Open **]**

7 References

SPT2CE-3532 - References

No	Title	Status
1	I2-Hardware Abstraction Interface Specification	In Review
2	I3 Interface Specification	In Review
3	System Analysis	Released https://rail-research.europa.eu/v1-release/
4	D26.3 – Final Modular Platform requirements, architecture and specification	Released: https://rail-research.europa.eu/wp-content/uploads/2025/04/D26.3-%E2%80%93-Final-Modular-Platform-requirements-architecture-and-specification.pdf

[Open **]**