# D1.2

# System Integration Requirements Document for Next-Generation TCMS

| | |
|---|---|
| **Project number:** | 730830 |
| **Project acronym:** | Safe4RAIL |
| **Project title:** | Safe4RAIL: SAFE architecture for Robust distributed Application Integration in rolling stock |
| **Start date of the project:** | 1st of October, 2016 |
| **Duration:** | 24 months |
| **Programme:** | H2020-S2RJU-OC-2016-01-2 |

| | |
|---|---|
| **Deliverable type:** | Report |
| **Deliverable reference number:** | ICT-730830 / D1.2 / 1.1 |
| **Work package** | WP 1 |
| **Due date:** | March 2017 – M06 |
| **Actual submission date:** | 31st of March 2017 |

| | |
|---|---|
| **Responsible organisation:** | TTT |
| **Editor:** | Mirko Jakovljevic |
| **Dissemination level:** | Public |
| **Revision:** | 1.1 |

| | |
|---|---|
| **Abstract:** | This document provides an initial set of system, functional and non-functional requirements related to system integration for the design of next-gen TCMS. WP1 approach to requirement collection is presented. |
| **Keywords:** | Integrated Modular Platform, System Integration, Requirements, Epics, Use Cases, Embedded Platform, Networking, Drive-by-Data |

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730830.

**Editor**

Mirko Jakovljevic (TTT)

**Contributors** (ordered according to beneficiary numbers)

Nataša Simanić-John, Derya Mete Saatci, Arjan Geven (TTT)

Bernd Löhr (NEW)

# Executive Summary

This document focuses on collection and analysis of system-level and embedded platform requirements, in relation to system integration and networking. It offers a foundation for a “Drive-by-Data” railway rolling stock backbone network, required for integrated modular architectures and next-generation TCMS, which can:

- host critical (up to SIL4) and non-critical functions in Ethernet networks based on strict temporal and spatial partitioning,
- support design of open and closed systems, i.e. including both predefined critical and other a-priori unknown network traffic for robust integration,
- enhance modularity and composability of embedded platforms and architectures from the networking perspective, thus reducing the complexity of system design, integration, reconfiguration, verification, certification and maintenance,
- lower the costs and effort of integration and certification for different subsystems and functions.

Being tied to SAFE4RAIL WP1 activities, this document describes:

- adopted requirements methodology,
- generic and basic Integrated Modular Platform functions and capabilities offered by the network,
- Provision of the Drive-By-Data network services offered to the the Functional Distribution Platform, i..e services that relate to the integration of network with the middleware services provided to applications:
    - message exchange
    - time synchronization services for the alignment of hosted functions and related to system integration,
- basic system integration requirements, functional and non-functional.

# Contents

# List of Figures

# List of Tables

# Chapter 1 Scope and Applicability

## 1.1 Requirements Methodology, Management and Ontology

The quality of requirements is measured according to how they are formulated. Especially in international project teams, it is highly important to follow a defined systematic approach in regard of the creation of requirements. In August 2016, the EuroSpec consortium released a new version of their document named “EuroSpec Requirement Management”. Besides featuring criteria for the creation and the check of the requirements quality, it outlines the importance of traceability, to make sure that all requirements made will be implemented in the system and can be proofed during the verification phase of the V-model and also explains ways on how to validate and verify requirements.

Depending on requirement’s intention and nature, this document will comply to the following requirements syntax (as defined by ISO 29148:2011 and the Easy Approach to Requirements Syntax (EARS)):

- universal: *The <system>* ***shall*** *<response>.*
- event-driven: *When <trigger>, the <system>* ***shall*** *<response>.*
- state-driven: *While <state>, the <system>* ***shall*** *<response>.*
- unwanted behaviour: *If <trigger>, then the <system>* ***shall*** *<response>.*
- optional: *Where <feature>, the <system>* ***shall*** *<system-response>.*

There is a ten criteria checklist for a requirement to satisfy:

1. Complete, clear, understandable sentence formulation
2. Active voice formulation
3. EARS syntax compliance
4. Absence of “weak words”:
    a. Comparatives/superlatives (e.g. better-best, smallest, largest)
    b. Subjective statements (e.g. easy, good, user-friendly)
    c. Ambiguous terms (e.g. optimal)
    d. Open-ended statements (e.g. at least)
    e. Loopholes (e.g. as applicable, possible)
    f. Negative statements (e.g. shall not)
    g. Connective statements (e.g. and, or)

h. Passive voice (e.g. shall be possible)

5. Description of a function or a property of a system/sub-system
6. Single requirement definition (no nested requirements, i.e. no multiple requirements within one definition)
7. Absence of implicit assumptions
8. Verifiable fulfilment
9. Defined version number and date
10. Completed attributes list (to be defined in later project phases).

EuroSpec recommends the use of the ReqIF format to interchange requirements (Chapter 7). A more detailed explanation on the format and how to use it can be found in D1.1 Workflow Methodology [1] deliverable from CONNECTA.

The EuroSpec document is freely available via the EuroSpec website.

The set of requirements for every topic follows the Requirements Ontology defined in the chapter 6.4.3 Requirements Attributes (Figure 6) of the D1.1 Workflow Methodology [1] deliverable from CONNECTA.

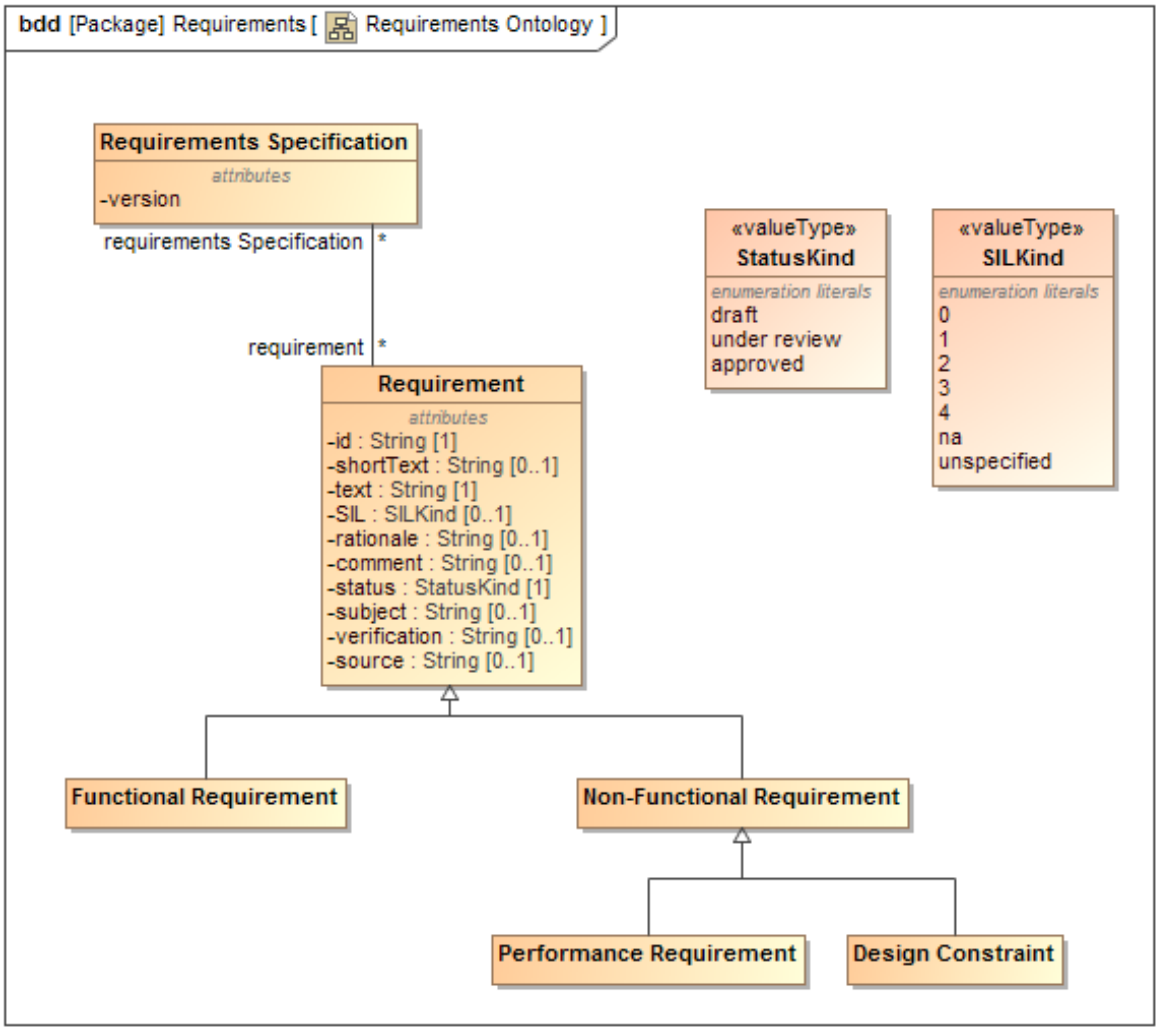


Figure 1 Requirements Ontology

The requirement collection process for integrated embedded platform consist of two basic parts – System Integration Requirements in WP1 “Drive-By-Data Networking”, and software platform and middleware requirements in WP2 “Functional Distribution Architecture Framework” as described in Figure 2. As the software platform focuses on computing modules and required software, middleware and RTOS with configuration, this document focuses on overarching concepts and requirements for system integration consisting of the networked system with many computers connected to the network.

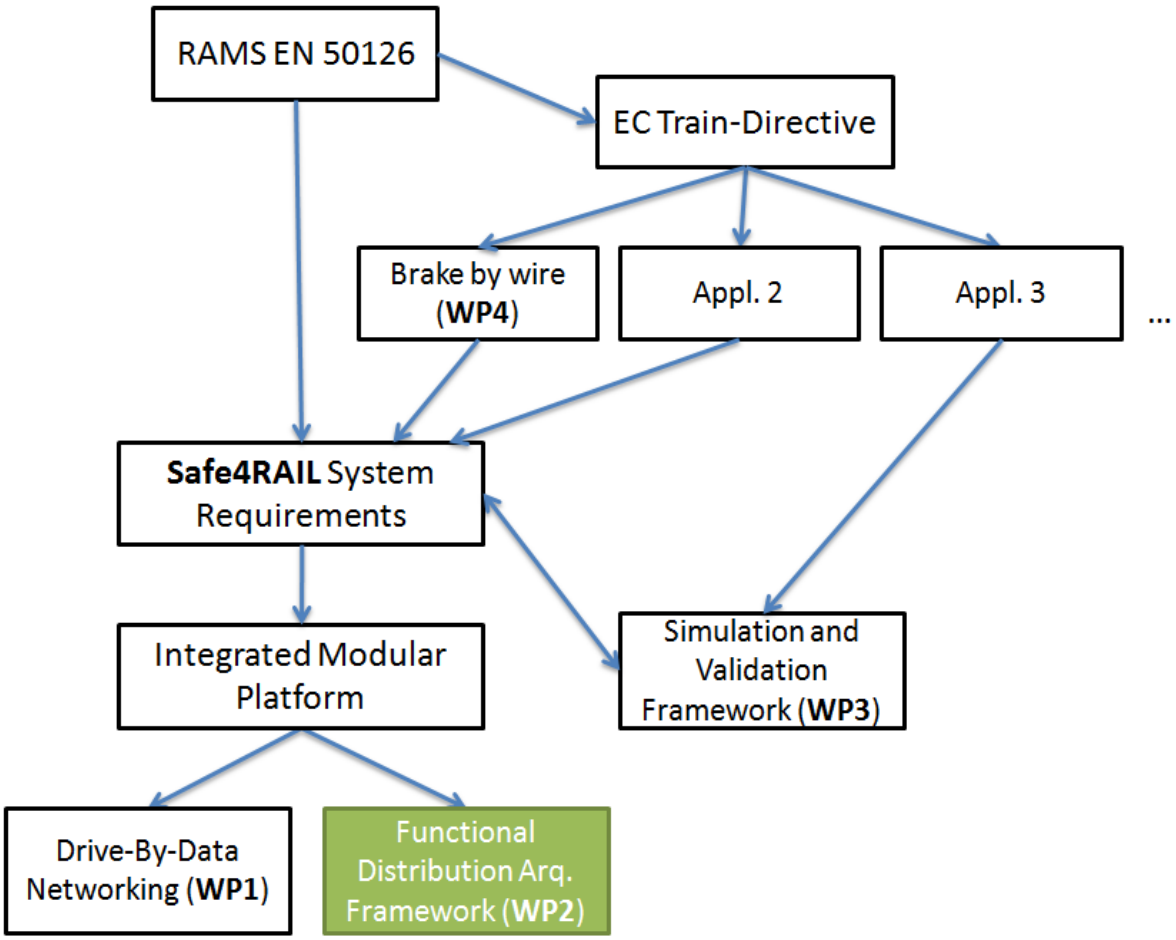


Figure 2 High-Level Definition of Requirements Process

## 1.2 Tailoring the requirement collection scheme

The EN 62559 describes a procedure for requirement management process and starts top-down with epics. The epics for S4R Integrated Modular Platforms (IMP) are listed as headline topics with very general and visionary objectives for IMP.

User stories are omitted as the user (human) will not experience anything of the IMP. The Drive-by-Data activities focus on generic platform capabilities, which can service application/function use cases on integration of trackside/signalling functions, safety lines, SIL4 functions, and other less critical functions. As a complementary information process differences related to IMP deployment may also be indicated, and as the experiences with such integrated system grow – user stories on the deployment will be updated.

Use cases typically describe the interactions of actors within the system in more details. However, the interfacing between functions and actors is configurable at IMP level and can be completely different formats and properties, depending on configuration, topology, hosted applications and system architecture. To better interpret functional requirements, possible platform states and transition activities contributing to the platform functions should be listed.

The requirements will be transformed to describe platform states with one or several functional requirements. The functional requirements will be collected in Task 1.1 and will be the basis for other WPs and tasks to specify the technical requirements.

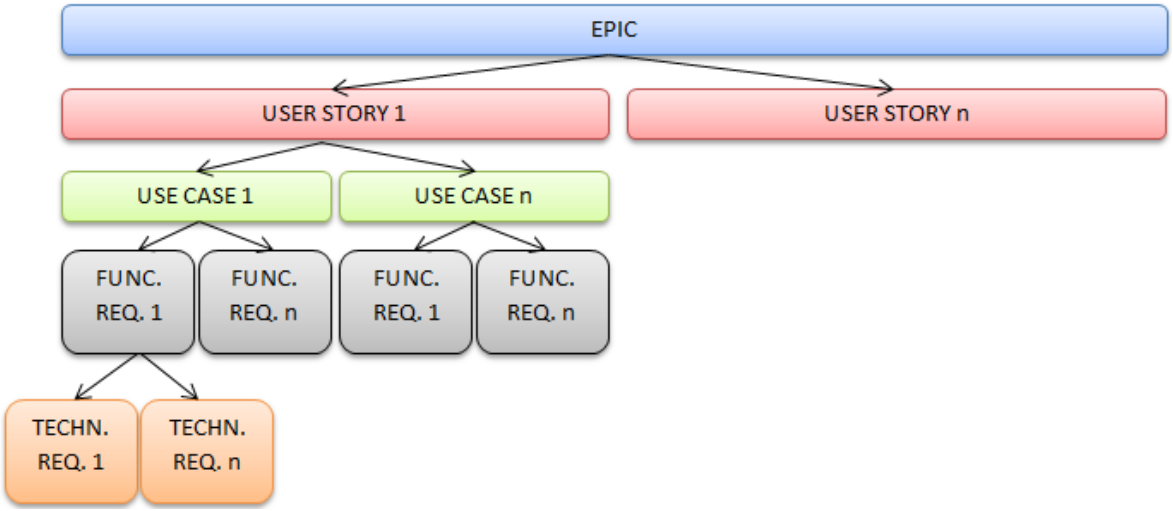


Figure 3 General Workflow Method applicable to TCMS and IMP

The approach presented in Figure 3 can be selectively applied to capture the requirements, assuming that the granularity and the level of requirements coming from different stakeholders are correctly interpreted.

### *1.2.1 Linking TCMS to IMP requirements*

While both TCMS and IMP can use a similar requirement capturing scheme, they will capture requirements on different levels; respectively, on train level and on subsystem level.

On this scale, obviously IMP requirements will be lower level to Train Operators, but its design and architecture may influence some of their concerns about the train capabilities (interoperability, availability, maintenance, upgrades, system obsolescence management, etc.). Other stakeholders such as OEMs and subsystem suppliers may have more defined use cases, application constraints, functional and technical requirements which will determine the capability of IMP for next gen-TCMS. In addition, not yet known and not well understood roles and process requirements (user stories) will be taken into account during the project, iteratively, as the level of understanding grows.

Finally, from TCMS user stories by train operators and from OEM's TCMS use cases, IMP epics will be defined. They will be supported by IMP use cases, which may be derived from OEM processes, railway regulations and guidance, and other cross-industry experiences.

### *1.2.2 Epics on Integrated Modular Platforms*

An epic in the context of requirement management for IMP is the description of requirements on a very high level of abstraction, providing a rough overview of the system requirements, without going into details.

In the case of SAFE4RAIL, an example could be the function of “simplify TCMS integration”, “hosting mixed-criticality TCMS functions” or “complexity reduction”.

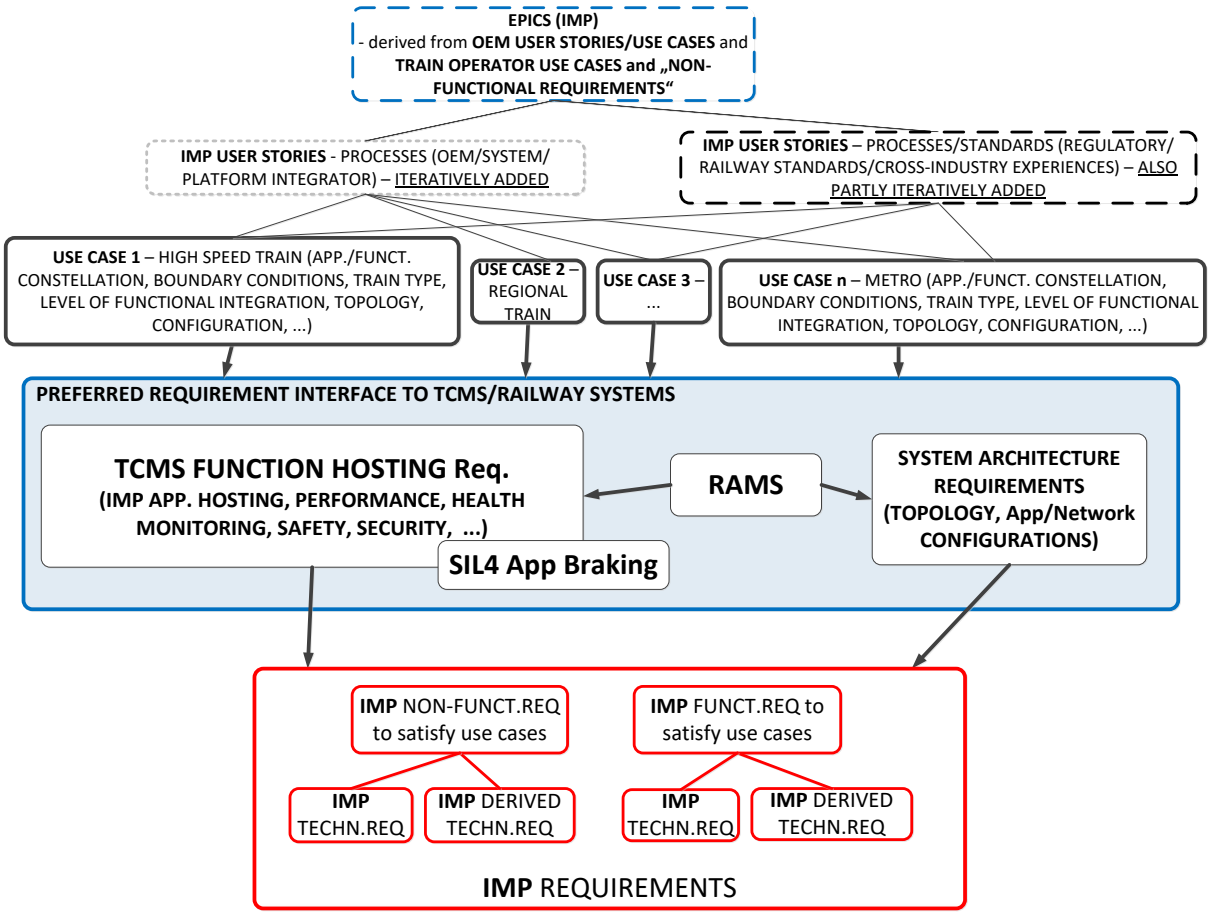


Figure 4 Workflow Method for IMP Requirements (simplified version)

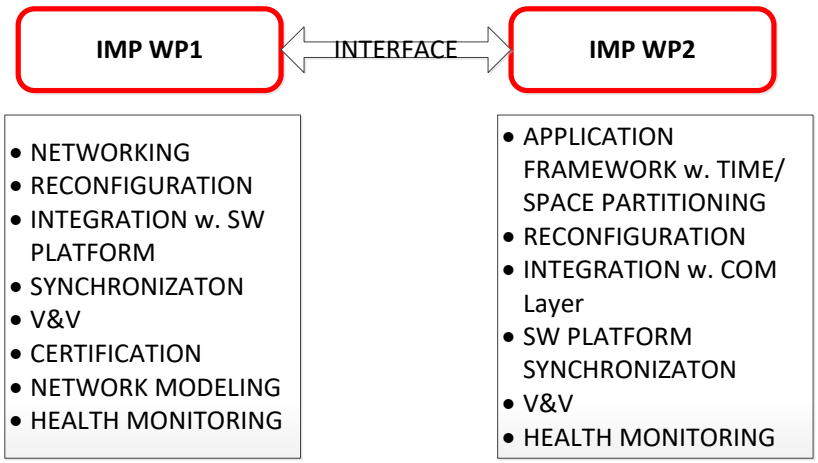


Figure 5 Separation of IMP Requirements to Drive-by-Data (WP1) and Functional Distribution Architecture (WP2)

### 1.2.3 IMP User Stories

IMP user stories will define the processes and stakeholders which participate in the IMP design, integration and certification process. Users groupings can be separate entities or within one company.

In the process of requirements collection, initial user stories will be described and later reviewed with stakeholders and regulatory authorities. By updating the initial set of user stories and processes for IMP deployment and incremental certification, a baseline for homologation and certification of next generation TCMS will be defined.

### 1.2.4 IMP Use Cases

Use cases for an integrated modular platform in the transportation industry can be generalized into different types of applications (high-speed train, regional train, tram, metro, etc). The solutions and mechanisms for every use case can slightly differ depending on target industry topology, failure hazard analyses, use case constraints and limitations on the vehicle use. The severity and criticality of specific rolling stock application and applicable national standards can contribute to further differentiation of use cases.

### 1.2.5 IMP Functional and Non-functional requirements

Functional requirements define specific behaviour or functions. Non-functional requirement specifies criteria that can be used to judge the operation of a system, rather than specific behaviours.

Functional requirements are usually in the form of "system shall do <requirement>", an individual action of part of the system. This can be in the sense of a black box approach or as description of an input-output process.

In contrast, non-functional requirements are in the form of "system shall be <requirement>", an overall property of the system as a whole and not a specific function.

In general, functional requirements are planned during the system or platform design, while non-functional requirements are relevant for the specific system architecture or configuration. According to nomenclature and approach from railway projects, functional requirements define what a system is supposed to do and non-functional requirements define how a system is supposed to be. It is important that we can define IMP functions and describe the minimum non-functional requirements, which can satisfy a set of TCMS architectures taking into account all thinkable TCMS use cases and scenarios, with relevant functional and non-functional requirements.

Non-functional requirements will describe platform properties. Non-functional requirements can be "quality attributes", “quality goals”, "quality of service requirements", "constraints", “capabilities” and "non-behavioural requirements". Qualities for non-functional requirements can be divided into two types of execution qualities. One covers, security, safety, availability, reliability and integrity, which are observable during the system operation. The another covers evolutionary system qualities, such as configurability, testability, maintainability, extensibility and scalability, which are embodied in the static structure of the software and hardware components, and modifiable by using a specific configuration which guides the use of system resources, computing/processing power and networking bandwidth.

Other examples of non-functional requirements include: Accessibility, Operability, Safety, Security, Testability, Usability, Robustness, Reliability, Quality, Compatibility, Certification, Efficiency, Fault tolerance, Interoperability, Maintainability, Network Topology, Modifiability. Interchangeability, etc.

In general, functional requirements are planned during the system or platform design, while non-functional requirements are relevant for the specific system architecture or configuration. According to nomenclature and approach from railway projects, functional requirements define what a system is supposed to do and non-functional requirements define how a system is supposed to be. It is important that we can define IMP functions and describe the minimum non-functional requirements, which can satisfy a set of TCMS architectures taking into account all thinkable TCMS use cases and scenarios, with relevant functional and non-functional requirements.

### *1.2.6 IMP Lower-level Technical Requirements*

At IMP level all functional and non-functional requirements are technical requirements. IMP does not provide any user relevant functions – hosted TCMS applications will take care. Use cases which are related to specific vehicle configurations, inauguration adaptations and can be handled by different IMP parameter configuration, or have some limitations in performance, can impose use-case specific IMP requirements, while the majority of use cases will be generic for any transportation industry.

Related to Drive-by-Data, the lower lvel technical requirements relate to:

- provision of the Drive-By-Data network services offered to the the Functional Distribution Platform, i..e services that relate to the integration of network with the middleware services provided to applications:
    - message exchange
    - time synchronization services for the alignment of hosted functions and related to system integration,

#### 1.2.6.1 IMP Derived and Non-Derived Technical requirements

Several requirements are based on assumed properties and capabilities of the system that may not be explicitly obvious to the customer/end user or may not be their direct concern and hence may not or cannot be provided to the subsystem owner directly. However, they pose great importance to the designer and the verifier of the subsystem, since they include essential assets towards the finalization of the product and assurance of the customer satisfaction. Therefore, these requirements are generated at the subsystem level (i.e. IMP level) owing to the domain expertise of the subsystem owner. Typically, such requirements emerge during the implementation, when a higher-level function is decomposed, or new subsystem functions are created. In the case of IMP, such assumed, or derived, technical requirements result from the analysis and allocation of technical requirements to the logical architecture. Derived requirements are identified in their source.

Derived requirements can be partly associated with different high-level safety, non-functional and functional requirements. While in some cases there may not seem a direct connection to higher level requirements, there is always an association with common intent of the higher level requirement. Therefore, the traceability should be established between higher level system requirements and lower level derived subsystem requirements. In some cases, derived subsystem requirements may require some changes and modifications to the system level requirements, as they contribute to better understanding of the system and related future modifications.

Just like any other requirement, the validation and verification of derived requirements is required at subsystem level. However, since derived requirements are not present on system level specifications (i.e. TCMS), but only appear in subsystem specifications (i.e. IMP), they are not required to be validated at system level. In case derived requirements oppose system level objectives and requirements, especially the safety requirements, they may be asked to participate in the system level validation process as well. Therefore, a rationale with justification of validity, correctness and completeness should be provided together with the derived requirements in the subsystem specification. In a later phase, higher level requirements may be formulated that relate to the derived requirements defined here and thus traceability can be established in a later phase.

Regardless of the source of the requirements, all requirements (whether derived or not) shall be assessed for safety and system impact on TCMS or consist/train functions. This way it is possible to capture inconsistencies and ambiguousness on the rationale, correctness, completeness or identification of derived and non-derived requirements early.

The importance of traceability is reflected in [EUROSPEC]:

- *Detectability: Requirements traceability supports the proof, that requirements or objectives are implemented and fulfilled in a system.*
- *Impact analysis: Requirements traceability supports impact analysis, for example by analysing the consequences for other requirements when changing a requirement (change management) or by analysing the influence between requirements.*
- *Identification of sophisticated requirements in specifications and system functions*

### *1.2.7 Internal Interfaces of IMP between Network and Middleware*

To simplify discussion on interfacing – the following chart is presented in Figure 6. The highest level IMP requirements will interface on time dissemination and layer alignments as well as on key mechanisms for data exchange among functions (Pt 1. in figure). For data/configuration loading and diagnostics the SW platform (Pt 2. /3. in figure) may require some specific data uploads and downloads. Configuration I/F (Pt 4. in figure) will require some alignment between the software platform (applications) and network configuration.

These interfaces are partially defined in this document and partially present in deliverable D2.5 " Next Generation TCMS Framework Requirements" of Safe4RAIL. The interface requirements will be united in the next revisions of these documents and will be handled as a separate document.

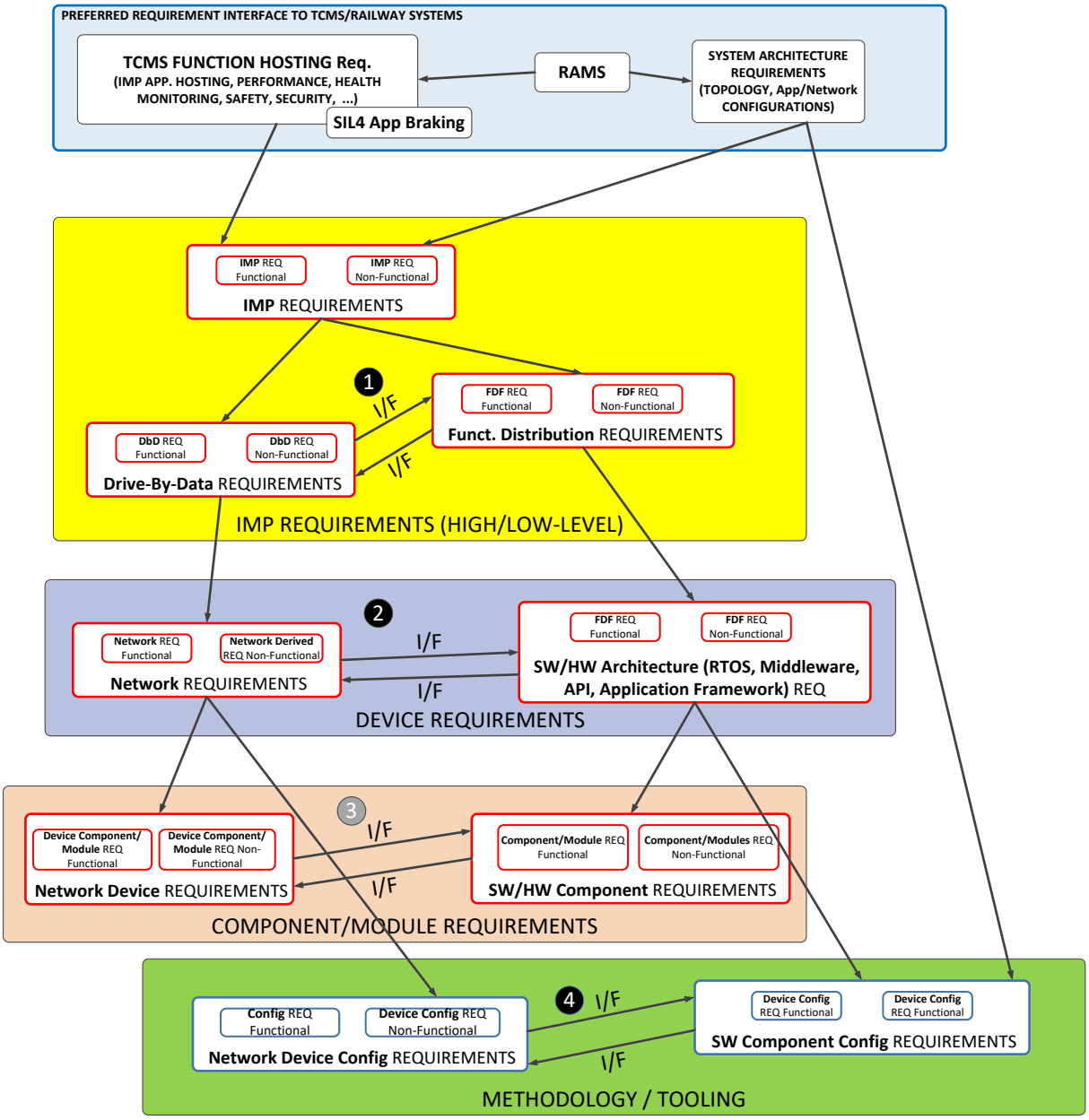


Figure 6 Interfacing Requirement definition for parameter-driven IMP

# Chapter 2 Integrated Modular Platform (IMP)

## Overview and Epics

The integrated modular platform hosts application functions and provides specific services to critical and non-critical applications, in order to establish robust software abstraction and provide all resources and timely information (sensors, global variables) access to applications.

Integrated modular platform is a part of the integrated system (see Figure 7, Figure 8). The configuration of integrated modular platform components, adapts the integrated modular platform to a specific use case and topology or architecture.

In the context of SAFE4RAIL, WP1 deals with Drive-By-Data and system integration baseline, while WP2 focuses on application hosting framework and software platform services. WP1 is concerned about anything related to system integration, interfacing and information transfer from one application partition to another application partition in the networked system. It focuses on all system integration capabilities required to define an integrated modular platform which can host different TCMS, door control, braking, safety or other non-critical functions in one system.

The Integrated Modular Platform does not depend on applications. Modular applications hosted on an Integrated Modular Platform can be tested in isolation and integrated on the system, without unintended interactions and interdependencies.

Therefore, an integrated platform can be seen as a subsystem, whose only function is to host different applications.

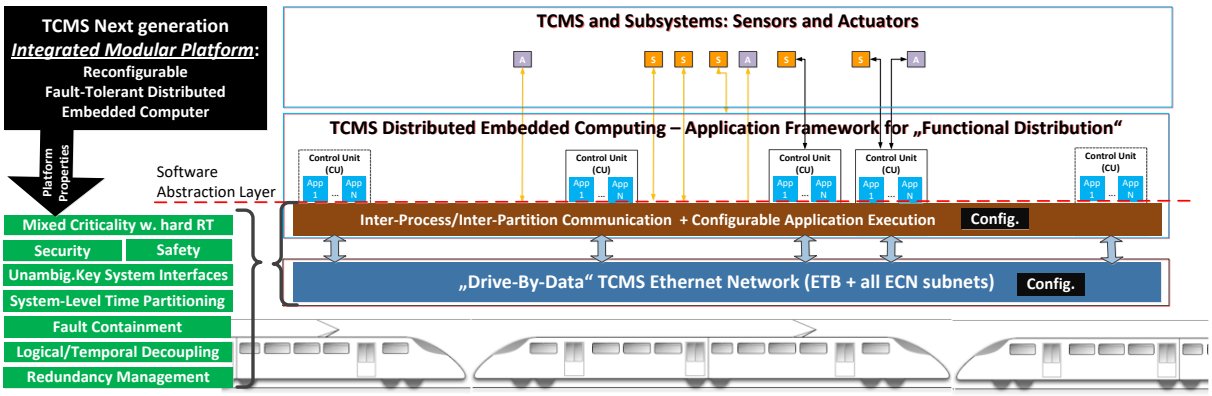


Figure 7 Integrated Modular Platform Layering

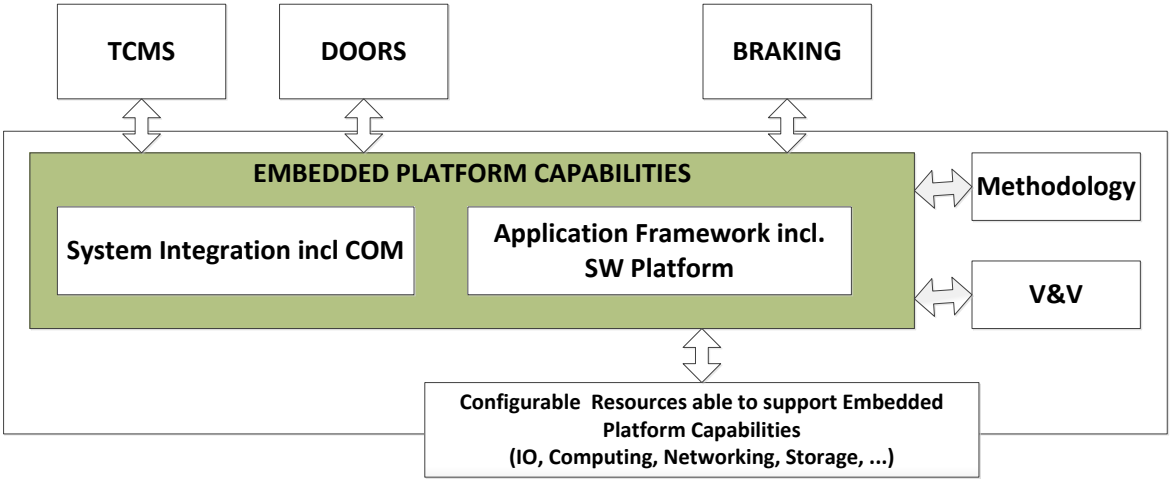


Figure 8 Embedded Platform and Requirements Grouping

## 2.1 Epics for next-gen TCMS Integrated Modular Platform (IMP)

The high level epics (see chapter 1.2.2) for the IMP give a description of requirements on a very high level of abstraction, providing a rough overview of the system requirements, without going into details.

For the IMP, the following eight epics have been defined:

**E1.** Next generation TCMS will **simplify integration** of a large number of functions on common computing and hardware resources to reduce system lifecycle costs for design, integration, V&V, testing, maintenance, upgrades, modifications, extension, incremental certification and modernization and reuse.

**E2**. Next generation TCMS will be **highly available and highly reliable** integrated platform, which will reduce physical system complexity, volume, number of connectors and decrease wiring length.

**E3**. Next generation TCMS will operate as a **fault-tolerant distributed computer** hosting all TCMS and other brake-by-wire, signalling, safety line, and non-critical applications.

**E4**. Next generation TCMS platform will be able to **integrate all critical and non-critical functions** relevant for train operation, including functional, performance, safety, security, availability and integrity requirements.

*Note: Next generation TCMS will support safe (SIL4) and secure operation (SL 3 and SL4).*

**E5**. Next generation TCMS will support **independent** design, testing, V&V and certification/homologation of functions.

*Note: Reuse, upgrades, extensions, incremental modernization and obsolescence management in next generation TCMS will be conducted without the need to retest and verify all already integrated functions.*

**E6**. Next generation TCMS platform will **establish and guarantee timing and performance of all critical functions**, based on system integration configuration.

**E7**. The IMP will support a **(re)configuration management system** that is robust and easy to maintain.

**E8**. The IMP will provide **interoperability** with respect to different and changing train configuration at functional and system integration level.

# Chapter 3 High-level system integration requirements for next-gen TCMS

### *3.1.1 Introduction*

This chapter serves as a baseline for more detailed system integration requirement discussions, and further structuring and scoping of requirements grouping and attributes. The set of requirements does not represent a complete, correct and unambiguous system integration requirements baseline, but represents an early analysis and expression of requirement considerations that will be refined until the end of the project.

### *3.1.2 Overview and Scope*

This chapter starts with very high-level system integration requirements and epics, and adds more detailed system integration requirements at software platform, network, and network device level, including the network configuration requirements.
The interface between the Drive-by-Data Network (WP1) and the Functional Distribution Architecture (WP2) is described in the following image. WP1 includes networking aspects and basic interfacing to the software platform (see Figure 9).

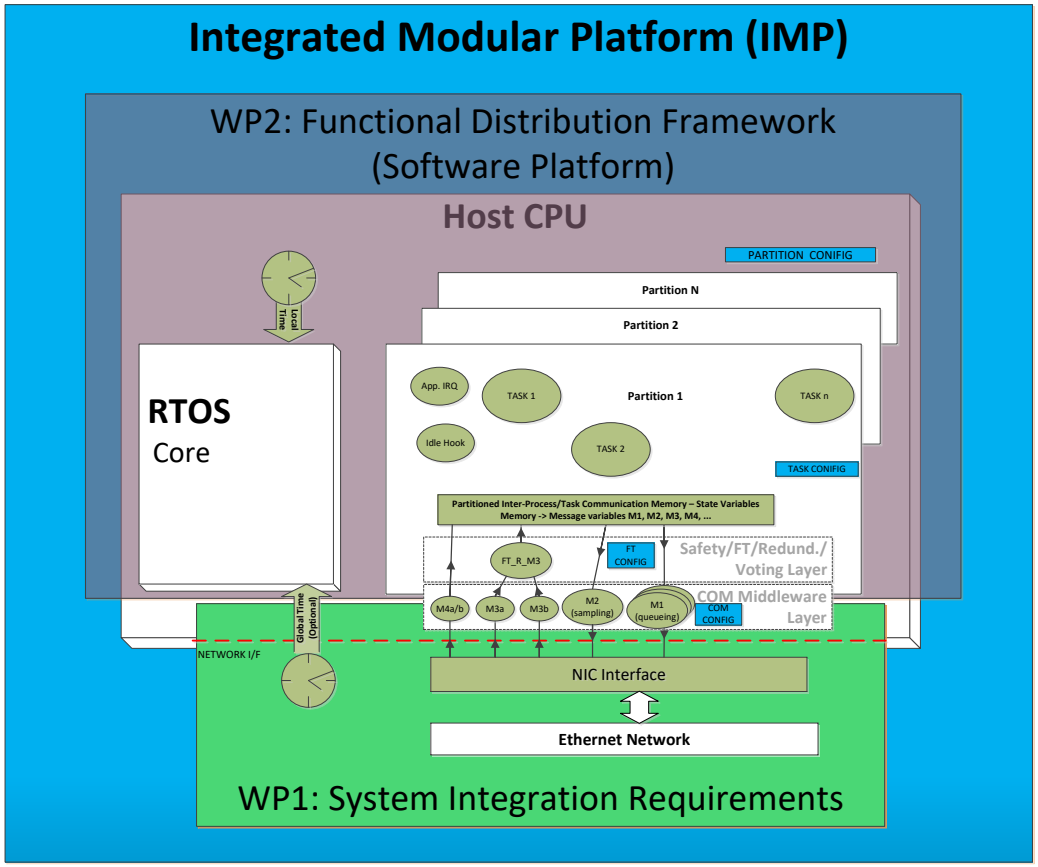


Figure 9 The scope of System Integration Requirements in WP1, related to WP2 (Logical View)

The system integration requirements can be split into traceable and derived requirements:

- Functional Requirements (from Ethernet networking integrated modular architectures, TCMS and use cases)
- Performance Requirements (also from TCMS and other use cases)
- Interface Requirements
- Safety Requirements
- Security Requirements
- Other RAMS Requirements

which can be traced to system-level and TCMS requirements and rationale, to other railway industry standards and cross-industry best practices on safety, security and design processes.

### *3.1.3 Integrated Modular Platform (IMP) - Scope and Capabilities*

This chapter provides assumptions on basic capabilities for SW platform. More detailed requirements can be found in WP2 deliverable D2.5 (planned release: April 2017).

Furthermore this chapter will be upgraded and outsourced to a separate document, after the alignment among WP1 and WP2 teams.

#### 3.1.3.1 Scope and interfacing system integration layer and software platform

Software platform consist of:

- RTOS scheduler and time/space partitioning services
- local clock and RTOS mechanism/drivers to align local clock to global time at defined periods
- time-aware schedulable task execution capability
- time-aware schedulable partition execution capability
- low-priority background (idle) task execution capability
- applications framework with configured time budgets, schedulable and configurable start and end instant (which should fit requirements of application hosting)
- mechanisms for local inter-task and inter-partition messaging at the host computer
- access to inter-task and inter-partition messaging between application hosted on end devices (EDs) within the networked system
- software abstraction (COM) layer which can be synchronized to local or (typically) global clock
- partitioned memory space for local and global data and state variables used by different partitions
- additional safety and redundancy management layers which could be synchronized to local or global clock
- Health monitoring and fault management for software platform, partitions and tasks

System integration layer consist of:

- Deterministic Ethernet network with configurable dataflows and QoS performance
- Health monitoring and fault management targeting network, system integration, logical links and dataflows
- Integration of dataflows via COM layer and publisher/subscriber services to distributed messages and data to applications hosted on different end-devices in the system
- Other inter-task/inter-partition communication mechanisms provided in the software platform which direct data to be available for specific tasks or to be picked by IO tasks.
- Configuration for defined networking layer and task to task latency within the system

System integration layer shall provide precisely defined and unambiguous configurable interfaces for information and state variable exchange among distributed functions in the networked system

#### 3.1.3.2 Outline: Integrated modular platform (IMP) capabilities

This section outlines IMP platform capabilities only. In the next refinements of the requirements, this section will be turned into a high-level set of IMP requirements.

Our current assumptions are:

1. Integrated modular platform will support safe and secure system operation.
2. Integrated modular platform will offer full partitioning, separation and virtualization for hosted functions of different criticality.

   *Note: Integrated modular platform shall support integration of hosted functions without any adverse and unintended interactions with other hosted functions.*
3. Integrated modular platform will support incremental and modular certification.

   *Note: This guarantees that already installed, verified and validated functions will not be influenced by newly added and modified hosted functions.*
4. Integrated modular platform will support logical and temporal separation of application properties.

   *Note: This is to facilitate reuse and upgrades.*
5. Applications hosted by the IMP will interact only by using defined interfaces.
6. Integrated modular platform will support preconfigured non-blocking resource sharing for all hosted critical applications.
7. Integrated modular platform will continuously detect its own faults and errors via built-in platform health monitoring, and test the IMP operation on start-up.
8. Integrated modular platform will provide a configurable platform fault management.
9. IMP fault management will provide timely and detailed information on the integrated modular platform health to hosted applications and system health management functions.

### *3.1.4 Functional Requirements*

#### 3.1.4.1 Definition of Key System Interfaces for Integrated Architectures

Key system interfaces represent the interfaces between core system functions which exchange global state variables, and have well defined temporal behavior.

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1defks_1 | System integration layer shall define key system interfaces between applications hosted in different software partitions installed on different computers. |
| 2 | WP1defks _2 | System integration layer shall allow the configuration of application constraints and models of computation and communication. |
| 3 | WP1defks _3 | System integration mechanisms shall define logical dataflows with specified determinism (periodicity, bounded latency and bounded jitter) and message order. |
| 4 | WP1defks _4 | System integration layer design shall ensure that configured key system interfaces maintain the designed temporal properties. |

#### 3.1.4.2 System Integration Virtualization

This section lists relevant system integration virtualization requirements:

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1nv_1 | System integration layer shall provide full partitioning, separation and network virtualization for dataflows of different time-criticality (soft time, real-time, hard real-time/time-aware, …). |
| 2 | WP1nv_2 | System integration layer shall support train reconfiguration (including consist/car rotation) without influencing temporal performance of critical functions. |

### 3.1.4.3 System integration resource isolation and partitioning

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1nip_1 | System integration layer shall support non-blocking and congestion-free communication for all critical and safety-relevant functions. |
| 2 | WP1nip_2 | System integration layer shall support weak and tight coupling among hosted functions.<br>Note: asynchronous and synchronous Ethernet packet-switching communication with different levels of alignment and synchronization to be allowed. |
| 3 | WP1nip_3 | System integration layer shall support domain separation, and data diodes (configurable unidirectional dataflows). |
| 4 | WP1nip_4 | System integration layer shall ensure full isolation of all configured dataflow. |
| 5 | WP1nip_5 | System integration layer shall provide health monitoring and configuration mechanism.<br>Note: This is to ensure maintainability, but also keep proper isolation and partitioning of the network bandwidth. |
| 6 | WP1nip_6 | System integration layer shall monitor and enforce configured message traffic and dataflows with defined maximum rate. |
| 7 | WP1nip_7 | System integration layer shall interface with publisher/subscriber mechanisms at COM layer. |
| 8 | WP1nip_8 | System integration layer and COM(communication abstraction middleware) shall ensure partitioning and isolations for data collection and storage on target computer. |
| 9 | WP1nip_9 | System integration layer shall provide TCP, UDP, raw MAC access points for communication of soft-time, event-driven or best effort applications and functions. |

#### 3.1.4.4 Synchronization and time dissemination

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1td_1 | System integration layer shall support a fault-tolerant timebase and time dissemination. |
| 2 | WP1td_2 | System integration layer shall disseminate time base to all end-devices. |
| 3 | WP1td_3 | System integration layer shall disseminate a fault-tolerant timebase on redundant networks. |
| 4 | WP1td_4 | System integration layer shall disseminate a fault-tolerant timebase on ETB and ECN networks. |
| 5 | WP1td_5 | System integration layer shall support a fault-tolerant timebase whose precision does not depend on train reconfiguration or consist rotation. |
| 6 | WP1td_6 | System integration layer shall support a fault-tolerant train-wide timebase with precision below 20µs over 64 hops. |
| 7 | WP1td_7 | System integration layer shall support a fault-tolerant timebase within the consist with precision below 10µs. |
| 8 | WP1td_8 | System integration layer shall compensate timing dissemination for variations in wiring length of 30m in consist from different vendors. . |
| 9 | WP1td_9 | The health monitoring mechanisms of the system integration layer shall detect increasing or varying latencies over wiring. |
| 10 | WP1td_10 | Synchronization start-up shall have defined maximum bound (e.g. in milliseconds) – at ETB (train), ECN (consist) and end station level. |
| 11 | WP1td_11 | Synchronization recovery at every end-station shall have defined duration (e.g. in milliseconds) – at ETB (train), ECN (consist) and end station level. |

#### 3.1.4.5 Communication Abstraction Layer Interfacing

The objective of this chapter is to define the interface between System integration layer and functional distribution framework (software platform). This interface can be designed in hardware only or in hardware and software, depending on the host interface capabilities of selected Ethernet networking devices and NIC.

The following requirements has been identified:

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1iffd_1 | Software platfrom interfacing with communication layer (COM) shall be based on message exchange over communication ports and completely abstracted from underlying network topology and application specifics.<br><br>(Note: Messages can contain one or more application relevant state variables, signals or alarms, …) |
| 2 | WP1iffd_2 | Message data transfer shall be atomic. |
| 3 | WP1iffd_3 | Message data transfer shall be driven solely by configuration parameters in connection-less mode. |
| 4 | WP1iffd_4 | System integration layer shall allow the transmission of UDP, TCP, raw MAC frames in line with system integration layer configuration. |
| 5 | WP1iffd_5 | Transmitting communication ports shall be tied to one unicast/multicast deterministic dataflow. |
| 6 | WP1iffd_6 | Receiving communication port shall be tied to one deterministic dataflow. |
| 7 | WP1iffd_7 | Receiving communication port shall be configurable to provide message data from only one of different redundant deterministic dataflow. |
| 8 | WP1iffd_8 | Data exchange for critical and safety-relevant function shall be established via unidirectional sampling and queueing ports. |
| 9 | WP1iffd_9 | Unidirectional sampling and queuing ports configured and used by one partition shall be accessible only to tasks hosted in the partition. |
| 10 | WP1iffd_10 | Sampling ports shall accept only one message and hold it until new message is written in the same field reserved for the sampling port message. |
| 11 | WP1iffd_11 | Queueing ports shall buffer messages as FIFO with defined maximum queue length. |

#### 3.1.4.6 System Integration Health Monitoring

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1sihm_1 | System integration layer shall provide health information on network and device operation and status. |
| 2 | WP1sihm_2 | System integration layer shall log information and statistics on dataflow status, and erroneous receptions and transmissions. |
| 3 | WP1sihm_3 | System integration layer shall support configurable transmission of dataflow, link, device and network health and status data, to end devices (ED). |
| 4 | WP1sihm_4 | System integration layer shall identify synchronization faults, and end device (ED) synchronization status. |

### *3.1.5 System Integration Performance*

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1sip_1 | System integration layer shall define more than 256 deterministic dataflows per ETB port. |
| 2 | WP1sip_2 | System integration layer shall enable ECN latencies of up to 1ms |
| 3 | WP1sip_3 | System integration layer shall enable ETB latencies of up to 10ms |
| 4 | WP1sip_4 | System integration layer shall enable bandwidth utilization of >90% when the data from different synchronous packet sources is assembled and sent over one physical link. |
| 5 | WP1sip_5 | System integration layer shall enable bandwidth utilization of >15% when the data from different asynchronous packet sources is assembled and sent over one physical link. |
| 6 | WP1sip_6 | System integration layer shall enable bandwidth utilization of >50% when the data from different synchronous packet sources is assembled and best effort traffic is sent together with synchronous traffic. |
| 7 | WP1sip_7 | System integration layer shall enable bandwidth utilization of >10% when dataflows from different asynchronous and synchronous packet sources is assembled and sent over one physical link. |
| 8 | WP1sip_8 | Network device power-up time shall be less than 1 second. |
| 9 | WP1sip_9 | Network device start-up on reset time shall be less than 0.3 second. |
| 10 | WP1sip_10 | Network line/link bandwidth for ETB and ECN networks shall be 1Gbps or higher. |

- 

### *3.1.6 Network topology and redundancy*

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1ntr_1 | System integration layer shall support flat topology and gateway-less communication among ETB and ECN networks.<br>*Note: To minimize messaging latency and jitter.* |
| 2 | WP1ntr_2 | System integration layer shall support redundant networking over independent networks. |
| | WP1ntr_3 | System integration layer shall support path redundancy in non-redundant networks. |
| 3 | WP1ntr_4 | System integration layer shall support relaxed timing constraints with defined upper latency boundaries at ETB layer.<br>Note: maximum latency will be defined within a defined range, depending on the number of network switch hops (e.g. 1-15, 16-31, 32-47, 48-63). |
| 4 | WP1ntr_5 | System integration layer shall support redundant transmission with zero fail-over for redundant paths or redundant networks. |
| 5 | WP1ntr_6 | System integration layer shall support and tolerate different ECN topologies and redundancy approaches. |
| 6 | WP1ntr_7 | System integration layer shall define common ETB topology and redundancy approach. |
| 7 | WP1ntr_8 | System integration layer shall support the integration of SIL4 functions via ETB backbone and dedicated ECN subnets. |
| 8 | WP1ntr_9 | System integration layer shall support frequent topology changes and reconfigurations. |
| 9 | WP1ntr_10 | System integration layer shall support Layer 2 train inauguration and topology detection mechanisms defined in IEC61375. |
| 10 | WP1ntr_11 | System integration layer shall support different scalable N-redundant system architectures. |

### *3.1.7 RAMS Requirements*

#### 3.1.7.1 Reliability, availability and safety

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1rams_1 | System integration layer shall support highly available communications. |
| 2 | WP1rams_2 | IMP shall maintain its operability when a single device of the system integration layer fails. |
| 3 | WP1rams_3 | System integration layer shall support highly reliable communications. |
| 4 | WP1rams_4 | System integration layer shall support high-integrity communications. |
| 5 | WP1rams_5 | System integration layer shall support safety-relevant communications with defined temporal boundaries. |

#### 3.1.7.2 Maintainability and Testability

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1rams-main_1 | System integration layer components shall support on-condition maintenance and predictive information about potential failures. |
| 2 | WP1rams-main_2 | On potential (predictive) failure detection, system integration layer shall provide information to system health monitoring. |
| 3 | WP1rams-main_3 | Exchange and repair of system integration layer modules and components shall be possible without testing, or adaptations. |
| 4 | WP1rams-main_4 | System integration layer shall detect unintended topology modification. *Note: resulting from maintenance and repair activities* |
| 5 | WP1rams-main_5 | Power-on system integration layer BIST shall identify all faulty components. |
| 6 | WP1rams-main_6 | System integration layer should support stepwise network operation and frame dissemination operation.<br>*Note: this is to support system verification and virtual coupling.* |

#### 3.1.7.3 Scalability, Modifiability, Interchangeability and Incremental Certification

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1rams-main_1 | System integration layer shall scale to up to 63 hops on ETB and a minimum of 4 hops at ECN level without gateways. |
| 2 | WP1rams-main_2 | System integration layer and topology shall support provable incremental modifications without affecting already integrated critical functions. |
| 3 | WP1rams-main_3 | System integration layer shall support full and degraded operational modes by using dissimilar Ethernet traffic classes. |
| 4 | WP1rams-main_4 | System integration layer shall support clean layering and software abstraction to software platform.<br>*Note: for full interchangeability of deployed networks and software.* |

### *3.1.8 System Integration Security*

This section lists relevant system integration security requirements:

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1sec_1 | System integration layer shall support design of solid isolated security zones with controlled dataflows and a subset of safety-critical functions. |
| 2 | WP1sec_2 | System integration layer shall support design of data diodes which cannot be changed by software reconfiguration.<br>Note: Diodes are unidirectional logical links and connections. |
| 3 | WP1sec_3 | System integration layer shall support partitioning mechanisms which cannot be modified or influenced by Denial-Of-Service attacks. |
| 4 | WP1sec_4 | Secure configuration data upload shall be supported. |
| 5 | WP1sec_5 | System integration layer shall prevent denial of service (DoS) attacks on critical and safety-relevant functions. |
| 6 | WP1sec_6 | System integration layer shall prevent eavesdropping/sniffing. |
| 7 | WP1sec_7 | System integration layer shall prevent frame header and data modifications for critical and safety-relevant functions. |
| 8 | WP1sec_8 | System integration layer shall prevent IP spoofing for critical functions. |
| 9 | WP1sec_9 | System integration layer shall prevent man-in-the-middle attack for critical and safety-relevant functions. |
| 10 | WP1sec_10 | System integration layer shall prevent resource high-jacking for critical and safety-relevant functions. |
| 11 | WP1sec_11 | System integration layer shall prevent application-layer attacks on safety-relevant system integration resources. |
| 12 | WP1sec_12 | System integration layer shall monitor and prevent potential security breaches. |

### *3.1.9 Compliance and Interoperability*

This section lists relevant compliance and interoperability requirements:

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1iop_1 | System integration layer shall provide capabilities equivalent or comparable to other advanced integrated architectures, *i.e. in ARINC664 and SAE AS6802.* |
| 2 | WP1iop_2 | System integration layer shall be compliant and interoperable with the latest 802.1Q standards. |
| 3 | WP1iop_3 | System integration layer shall be compliant and interoperable with the 1000BASE-TX and 1000BASE-T1.<br>*Note: latest 802.3 standards* |
| 4 | WP1iop_4 | System integration layer shall be compliant and interoperable with SDN (Software Defined Networking). |

### *3.1.10 Configuration Management for system integration and integrated modular platform*

| N° | ID | Requirement-text |
|---|---|---|
| 1 | WP1sicm_1 | The IMP Configuration Management System (CMS) shall use the configuration data to activate or deactivate system integration modules, resources or functions. |
| 2 | WP1sicm _2 | System The IMP CMS shall enable dynamic and static configuration control and compliance check-up of the IMP, system modules, resources and applications |
| 3 | WP1sicm _3 | The IMP CMS shall define the parameters that affect the integrated system (schedule, resource reservation, performance, module part numbers). |
| 4 | WP1sicm _4 | The IMP CMS shall determine the allocation of the system integration resources to the hosted applications. |
| 5 | WP1sicm _5 | THE IMP CMS shall know about:<br>a) Hardware part numbers and serial numbers installed in the system<br>b) Hardware modification status indicators<br>c) Identity of all software part numbers installed in the system<br>d) Identity of all configuration data installed in the system<br>e) Identity of all database files installed in the system |
| 6 | WP1sicm _6 | IMP CMS shall be able to operate statically and/or dynamically.<br><br>Note: Statically - defined at design time, dynamically configured system can be adapted during TCMS operation |
| 7 | WP1sicm _7 | System Integration layer shall provide means for verifying the configuration and interoperability of network modules and components. |

### *3.1.11 Definitions*

**Train** IEC 61375-1 defines a train as a composition of closed trains and consists, each consist 0having one or several vehicles (cars), and each closed train having one or several consists.

**Consist** Usually the smallest network part, a group of coupled cars not separated or changed during normal operation.

**White Channel** A communication channel has well understood properties relevant for safety applications, and consists of devices and communication protocols designed to specific safety assurance/integrity levels.

**Black Channel** A communication channel without any known safety properties and capabilities. Middleware and functional application assumes it cannot rely on communication network for any safety-relevant activity, and provides mechanisms for the identification of communication faults.

**System Integration Layer** Networking and coupling with software platform layer which facilitates deterministic transmission of messages among computers hosting software applications. System integration layer shall provide precisely defined and unambiguous configurable interfaces for information and state variable exchange among distributed functions in the networked system

**Key System Interface** is the interface between core system functions which exchange global state variables over the system integration layer

**Network Virtualization** Network management and configuration which strictly separate, isolate and reserve network resources for specific functions.

**Partitioning** Allocation of the network resources dedicated to specific functions

**Separation / Isolation** Preventing unintentional or intentional use of network resources by functions which should not have access to those resources

# Chapter 4 Summary and conclusion

System integration represents a core capability required for the design of advanced integrated architectures, especially in *distributed* integrated architectures, which pose challenges to the transport and availability of data at different nodes in the system.

With advanced integrated systems tailored to host many critical and non-critical functions, the system integration gains in importance as it represents a common shared resource relevant for all functions. This document provides the initial set of requirements to the design of such a future distributed integrated architecture for the railway domain, referred to as “Integrated Modular Platform”.

The requirements that have been presented in this deliverable are split into several parts. A first part covers overall platform requirements for system integration purposes, such as those related to resource isolation and partitioning, scalability, and composability. A second part covers requirements targeted at the networking layer specifically, responsible to (timely) distribute data between the different nodes in the distributed system. Integration requirements related to the connection between the network and the middleware that hosts applications are described.

This documents provides an overview of the requirement collection process, requirement groupings and key stakeholders. Epics, IMP platform assumptions and WP1 scoping have been described for initial discussions and further review by key stakeholders. The objective is to create a baseline for further analyses and detailed collection of all system integration and networking requirements in WP1 of Safe4Rail, and to interface with requirements collected in WP2.

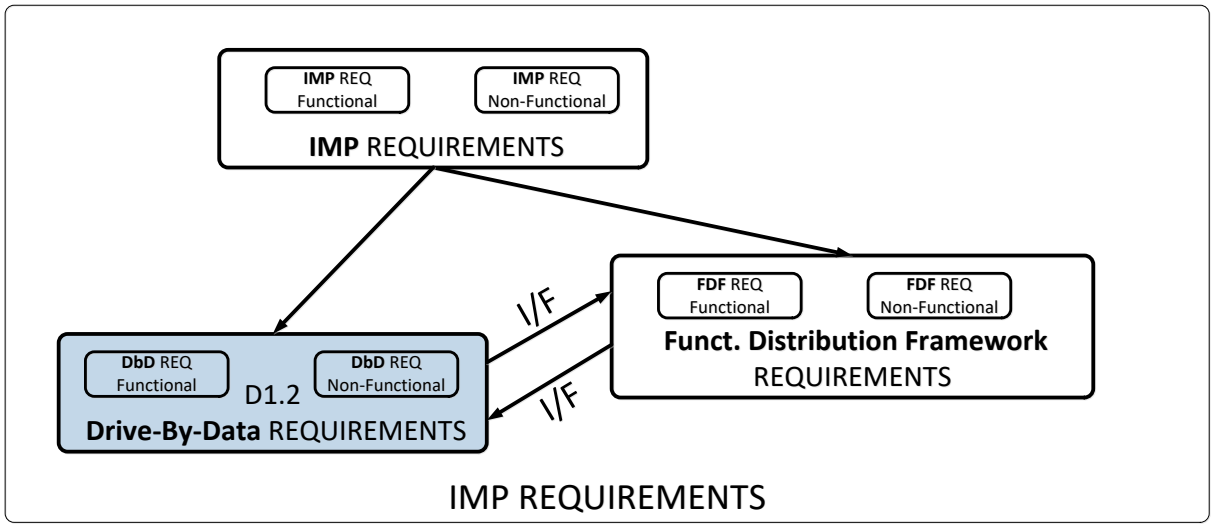


Figure 10 Requirements Traceability

The document will be in later versions be expanded to cover detailed requirements for definition of Drive-By-Data architectures for the next generation of TCMS. By reviews and incremental progression, this document will be matured and significantly expanded to accommodate OEM program needs.

# Chapter 5 List of Abbreviations

| | |
|---|---|
| **BIST** | Built-In-Self-Test |
| **CMS** | Configuration Management System |
| **COM** | COMmunication |
| **COM/MON** | COMmander/MONitor |
| **CPU** | Central Processing Unit |
| **CRC** | Cyclic Redundancy Check |
| **DHCS** | Data Handling and Communication System |
| **DoS** | Denial of Service |
| **ECN** | Ethernet Consist Networks |
| **ETB** | Ethernet Train Backbones |
| **FIFO** | First-In-First-Out |
| **IMP** | Integrated Modular Platform |
| **MTBF** | Mean Time Between Failures |
| **NIC** | Network Interface Card |
| **OEM** | Original Equipment Manufacturer |
| **RAMS** | Reliability, Availability, Maintainability, Safety |
| **RTOS** | Real Time Operating Systems |
| **S4R** | Safe4RAIL |
| **SDN** | Software Defined Networking |
| **SDT** | End-to-end protocol over an untrusted communication channel |
| **SIL** | Safety Integrity Level |
| **SL** | Security Level |
| **SW** | Software |
| **TCMS** | Train Control and Management System |

| | |
|---|---|
| **TCP** | Transmission Control Protocol |
| **TRDP** | Train Real Time Data Protocol |
| **UDP** | User Datagram Protocol |
| **VL** | Virtual Link |
| **WP** | Work Package |

Table 1: List of Abbreviations

# Chapter 6 Bibliography

[1] A. o. T. O. C. (. D. B. SNCF (SNCF), “Specification for the documentation of railway,” EU Railway Operators, 2016.

[2] CONNECTA Deliverable D1.1, "Workflow Methodology, Dec 2016

[3] EuroSpec Requirements Management,from:
http://www.raildeliverygroup.com/files/Publications/services/eurospec/EuroSpecRequirementsManagementv2.pdf