OCORA

Open CCS On-board Reference Architecture

System Capabilities

Document ID: OCORA-TWS01-020

Version: 1.00

Release: R1

Date: 03.12.2021

Revision history

Version	Change Description	Initial	Date of change
1.00	Official version for OCORA Release R1	MT, JL, PV, TM, AL	03.12.2021

Table of contents

Table of figures

Table of tables

References

Reader's note: please be aware that the numbers in square brackets, e.g. [1], as per the list of referenced documents below, is used throughout this document to indicate the references to external documents. Wherever a reference to a TSI-CCS SUBSET is used, the SUBSET is referenced directly (e.g. SUBSET-026). OCORA always reference to the latest available official version of the SUBSET, unless indicated differently.

[1] OCORA-BWS01-010 – Release Notes

[2] OCORA-BWS01-020 – Glossary

[3] OCORA-BWS01-030 – Question and Answers

[4] OCORA-BWS01-040 – Feedback Form

[5] OCORA-BWS03-010 – Introduction to OCORA

[6] OCORA-BWS04-010 – Problem Statements

[7] ISBN 978-1-78548-169-7 – Model-based System and Architecture Engineering with the Arcadia Method – Jean Luc Voirin – ISTE Press - 01/03/2018

1 Introduction

1.1 Purpose of the document

The purpose of this document is to provide a high-level view of all services that an OCORA based system provides to external actors, and to prepare for Europe's Rail Joint Undertakings System- & Innovation-Pillar. The document will evolve in subsequent releases. At a later stage modelling of system capabilities is expected to be performed using a modelling tool such as Capella.

This document is addressed to experts in the CCS domain and to any other person, interested in the OCORA concepts for on-board CCS. The reader is invited to provide feedback to the OCORA collaboration and can, therefore, engage in shaping OCORA. Feedback to this document and to any other OCORA documentation can be given by using the feedback form [4].

1.2 Applicability of the document

The document is currently considered informative. Subsequent releases of this document will be developed based on a modular and iterative approach, evolving within the progress of the OCORA collaboration.

1.3 Context of the document

This document is published as part of the OCORA release R1, together with the documents listed in the release notes [1]. Before reading this document, it is recommended to read the Release Notes [1]. If you are interested in the context and the motivation that drives OCORA we recommend to read the Introduction to OCORA [5], and the Problem Statements [6]. The reader should also be aware of the Glossary [2] and the Question and Answers [3].

2 Definition of System Capabilities

2.1 System Capabilities in context of the Arcadia method

Ever-increasing expectations regarding functionality, safety, security, and performance of today's railway CCS systems, originating from different stakeholders, need to be managed and implemented in a structured and traceable way. The strength of any system is rooted in its architecture. The Arcadia method supports all required engineering activities that include analysing operational needs, structuring, and decomposing the system and considering constraints of existing standards and legislation as well as domain specific design objectives.

System Capabilities are a core element of the Arcadia methodology and defined [7] as follows:

"*A system capability is the system's expected ability to supply a service contributing to fulfilling one or more missions. A system capability represents a system usage context. It is characterized by a set of functional chains and scenarios that it references, and which more precisely describe the conditions for performing the system functions that contribute to it. A capability can also reference a function that contributes to it by itself. A capability can use one or more other capabilities that it will reference.*"

2.2 OCORA interpretation

For a better understanding, especially for people who are not familiar with the Arcadia method, capabilities may be considered services of the system provided to external actors. Capabilities are characterized, described, and illustrated using functions, scenarios, modes, and states.

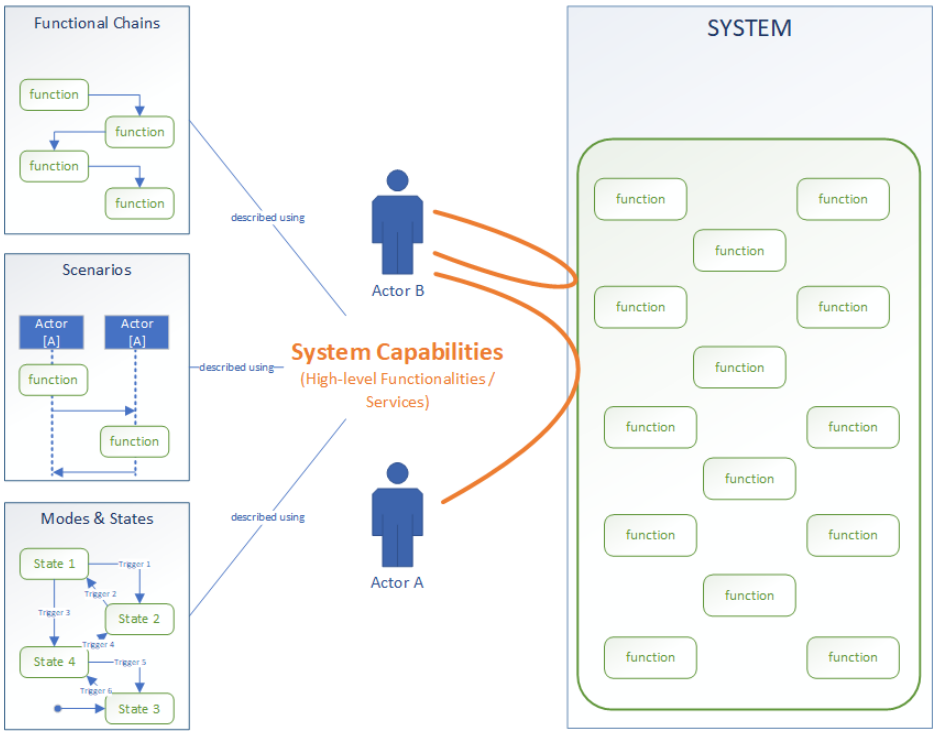


Figure 1 OCORA Perspective of System Capabilities

2.3 Arcadia approach

Arcadia stipulates a viewpoint-driven approach (as described in ISO/IEC 42010) and emphasizes a clear separation of need and solution.

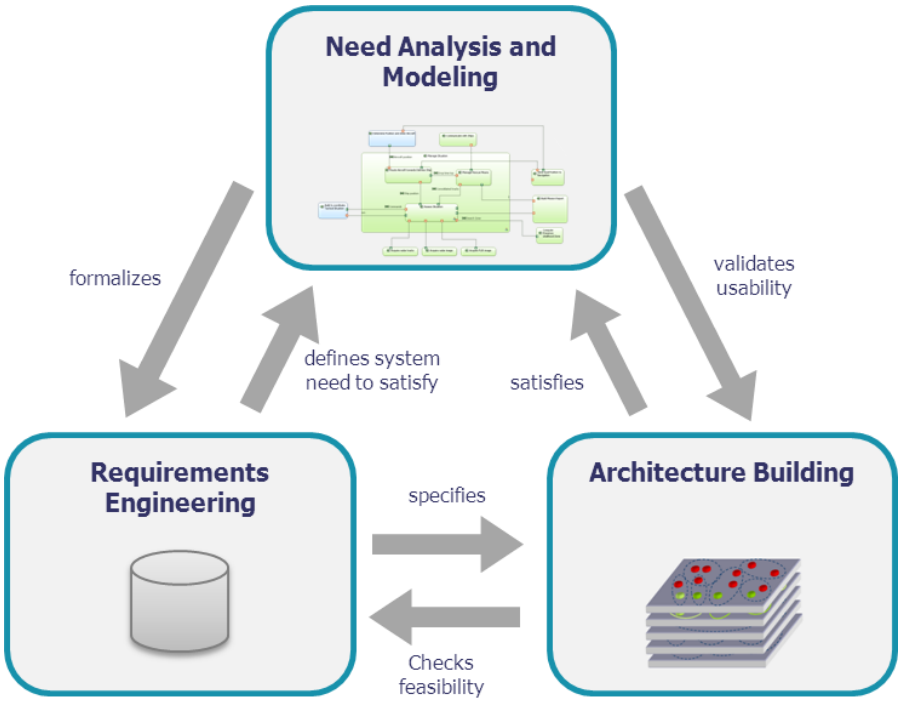


Figure 2 Viewpoint driven approach[1]

Arcadia promotes four[2] distinct perspectives:

- **Customer Operational Need Analysis** - definition of the Problem
  Focuses on analysing the customer needs and goals, expected missions and activities. It structures the need in terms of actors/users, their operational capabilities, and activities.

  Note: *Linx4Rail conducts an overall Operational Analysis. Synchronization shall happen at a later stage.*

- **System Need Analysis** - formalization of system requirements
  Focuses on the System itself, to define how it will satisfy the compiled operational need - zeroing in on functions and its related exchanges, non-functional constraints (e.g. safety, security, etc.) as well as role sharing between system and actors.

- **Logical Architecture (Notional Solution)** - definition of solution architecture
  Aims at building a coarse-grained component breakdown of the system. This involves taking important engineering decisions which are unlikely to be challenged at a later stage. The system is decomposed into logical components, functions are allocated to components. This building process is where the majority of the OCORA design objectives and design rules will be considered.

- **Physical Architecture** - definition of solution architecture
  Makes the logical architecture vision evolve according to implementation, technical and technological constraints, and choices.

[1] Source : https://www.eclipse.org/capella/arcadia.html
[2] Arcadia includes a fifth perspective regarding the Building Strategy. However, this perspective is currently considered to be out of scope of OCORA.

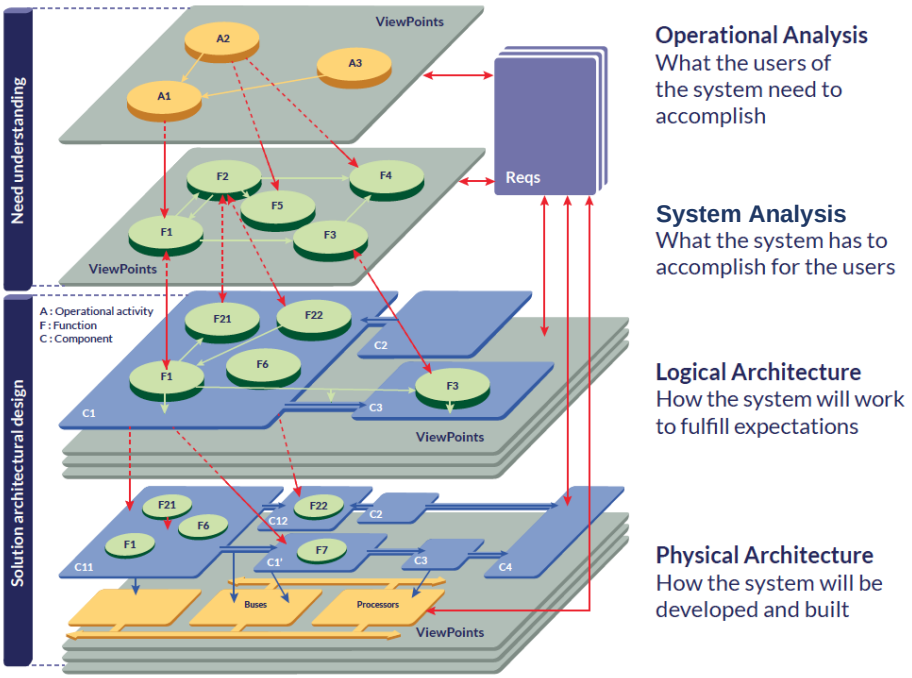


Figure 3 The four perspectives of Arcadia1

The Arcadia approach has been synchronized with the OCORA tailored V-model development process (see also Requirement - Management Guideline [9]).

Important Arcadia artefacts are:

Artefact	Description
System	The system is an ordered set of elements functioning as a whole, responding to customer and user demand and needs, and subject to engineering supported by Arcadia.
Actor	An actor is an entity that is external to the system (human or not), interacting with it, especially via its interfaces.
Mission	A mission is a high-level goal to which the system should contribute. To be fulfilled, a mission should use a number of system functions, regrouped within one or more system capabilities.
Capability	A system capability is the system's expected ability to supply a service contributing to fulfilling one or more missions.
Scenario	A function scenario is a time-ordered dynamic flow, on a temporal axis (conventionally vertical from top to bottom), of exchanges between different functions in the context of implementing a capability.
Functional Chain	A functional chain is an ordered set of references to functions and the functional exchanges that link them, describing one possible path among all the paths forming the dataflow.
Mode	A mode is a behaviour expected of the system, a component or also an actor or operational entity, in some chosen conditions.
State	A state is a behaviour undergone by the system, a component, an actor or an operational entity, in some conditions imposed by the environment.
Mode/state diagram	A mode(s) machine (or respectively, state(s) machine) is a set of modes (or, respectively, states) linked to one another by transitions. Modes and states cannot cohabit in the same machine.

Table 1 Arcadia Artefacts explained

[1] Source : https://www.eclipse.org/capella/arcadia.html

The following figure shows the relationship between Capabilities and other Arcadia modelling artefacts:

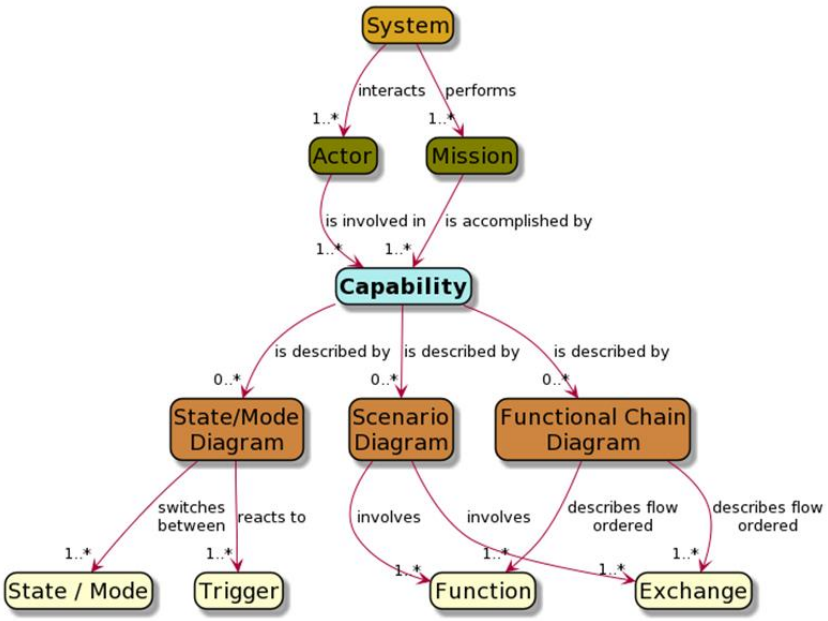


Figure 4 Artefacts of the System Analysis perspective

3 Operational context

To facilitate the identification of System Capabilities of the CCS-OB, OCORA identifies the need for a high-level operational context of the different types of trains. The goal of this process is to help identifying the System Missions and high-level functional interactions of all actors of the CCS-OB.

3.1 Overview

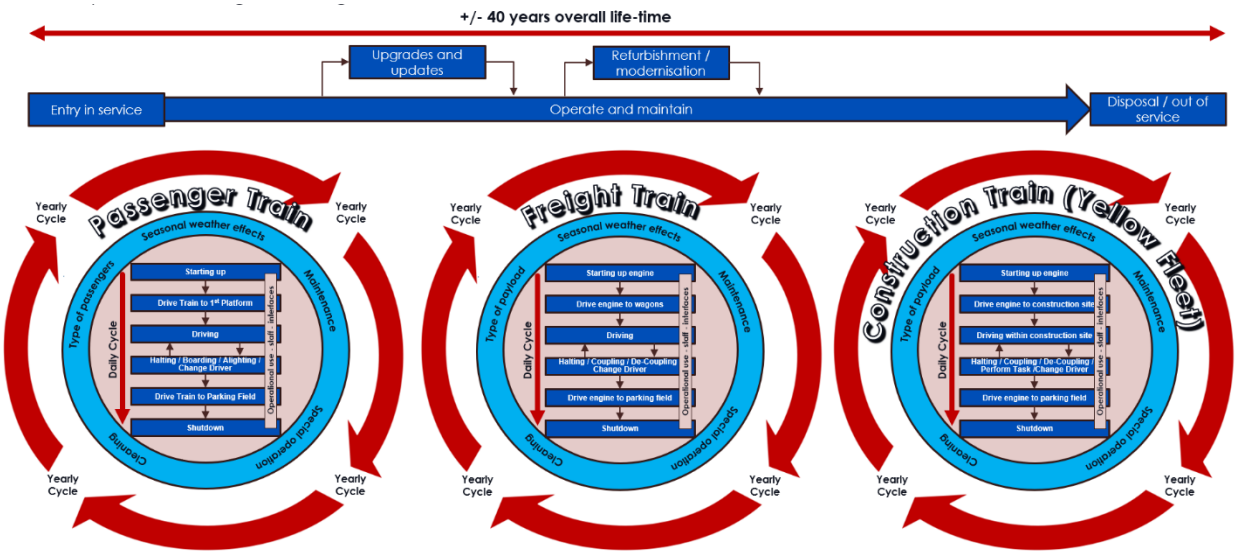


Figure 5 Operational concept overview

The high-level operational concept describes the production model of the operator: operational processes with a focus on rolling stock. The objective is to facilitate clear communication during the different project phases and provides context for expectations and user perspectives behind the formalised requirements. Within the V-model, the operational context is located at both upper ends of the V: it is an information source used when technical system requirements are written, but also the basis for validation activities. Considering the train within the context of operation, a framework of three different scales is central to this:

- A **day cycle**. The daily cycle will cover the daily operational processes, from preparing the train for passenger service until the end of service.
- A **yearly cycle.** The yearly cycles will zoom out further, assessing the weekly, monthly and annual processes. This can cover both maintenance processes, atypical operational procedures, but also the influence of seasonal effects.
- The full **life cycle** from introduction up to and including the end of service and recycling. As the life cycle of the train will contain multiple life cycles of different IT systems, this perspective is the basis to prepare for regular updates and upgrades, ideally in sync with the longer maintenance cycles.

Combining the need different time frames, which will include a set of operational processes, serves as a first perspective to determine required capabilities of the CCS_OB, which is defined further in chapter 5.

4 System analysis

Once the System under consideration is defined (scope and boundaries), the most natural way to compile the full set of System Capabilities is to follow a top-down approach and start by identifying Actors interacting with the System and Missions the System shall perform (see 4.1 System scope and actors and 4.2 Missions).

Knowing the Missions of a System is prerequisite to identify the System Capabilities necessary to accomplish those missions (see 4.3 System Capabilities). Again, Capabilities facilitate the identification of System Functions. Capabilities are described using one or multiple Functional Chain(s), Scenario(s) and Mode/State Diagram(s).

Functional Chains assist the RAMSS analysis, Scenarios describe the sequence of functions and their data exchanged. All resulting artefacts support the verification and validation process.

4.1 System scope and actors

As a first step for defining the Missions and System Capabilities, the scope and corresponding actors are identified.

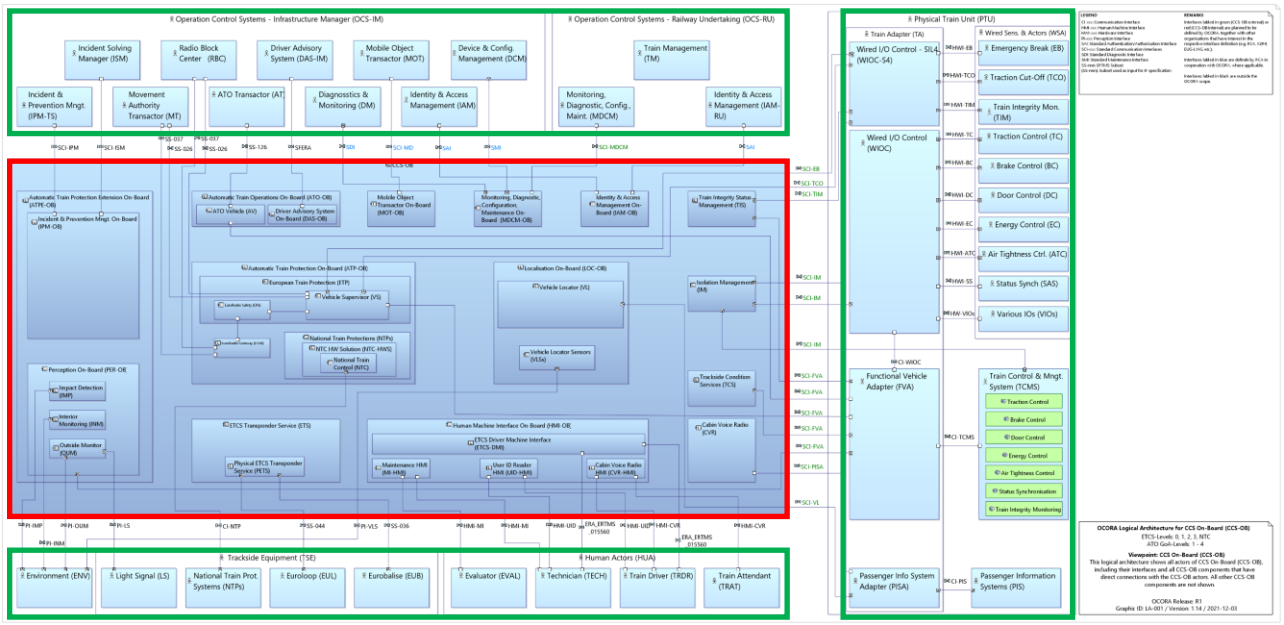


Figure 6 System under consideration (red) and Actors (green)

Note: The System under Consideration (SuC) is CCS On-Board only e.g., excluding the Train Adaptor. Nevertheless, OCORA aims to standardize the interface to the Physical Train Unit by leveraging existing standards and proposing improvements where necessary. Legacy trains require a Train Adaptor that translates legacy interfaces and behaviour; future trains, however, will implement the standardized interfaces natively, and will no longer need a Train Adaptor.

4.2 Missions

OCORA has identified the following missions the CCS-OB has to fulfil.

Ref.	Mission	ERTMS Level	GoA [1]	Comments
Mission 1	Control safe train movement.	0 1 2 3 NTC	1,2,3,4 1,2,3,4 1,2,3,4 1,2,3,4 1,2,3,4	This mission is related to Automatic Train Protection (ATP). Supervise safe train movement according to train configuration, track configuration, speed profile, permitted travel distance, assigned mode, current speed, and current position. This mission includes the triggering of EB and TCO.
Mission 2	Optimise (energy, capacity, comfort) train movement (command train or signal to driver).	0 1 2 3 NTC	1,2,3,4 1,2,3,4 1,2,3,4 1,2,3,4 1,2,3,4	This mission is related to Automatic Train Operation (ATO). Command train movement and release door opening according to train configuration, segment profile (topology), journey profile (schedule), mission profile (?), perceived environment information (e.g. clearance profile), speed profile, permitted travel distance, assigned GoA level, current speed, and current position (GoA2-4). Signal information to the driver about actual speed and the optimal speed profile (GoA1 only).
Mission 3	Signal information to ensure safe and appropriate driving.	0 1 2 3 NTC	1,2 1,2 1,2 1,2 1,2	This mission is related to ATP Cabin Signalling. Signal the mode of the ATP System and provide information to the driver about actual speed, maximum speed profile and the position within the permitted travel distance.
Mission 4	Provide safe voice communication between trackside and train.	0 1 2 3 NTC	1,2,3,4 1,2,3,4 1,2,3,4 1,2,3,4 1,2,3,4	This mission is related to Cabin Voice. The OCS-IM operator and the Driver (or Train Attendant in case of GoA3 [2]) can communicate safely with each other. In GoA4 the communication with passenger is bidirectional [3].

Remark: The maintenance of the system under consideration (CCS-OB) is not considered to be a mission of the system. For Level 0, Level 1 and Level NTC a system (e.g. perception, Euro-Loop, TVM, LZB) capable to provide track side signal information is needed to run in GoA3 and GoA4.

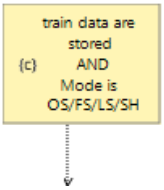
[1] The applicability of GoA with the ERTMS level needs to be verified by the ATO team (OCORA internal GitHub issue #259)
[2] Needs to be verified by the ATO team (OCORA internal GitHub issue #259)
[3] Needs to be verified by the ATO team (OCORA internal GitHub issue #259)

4.3 System Capabilities

This chapter lists and describes, on a high level, the capabilities applicable to CCS On-Board. At this point in time only a template with an example is provided. In subsequent releases more capabilities will be identified and described. For a detailed description of the capabilities, Capella will be used.

4.3.1 Capability description template

OCORA is planning to use the following template for identifying and describing, on a high level, the capabilities. Mandatory fields are marked with an (M) and attributes also foreseen in Capella are identified with an *.

Name*	(M)	Name of the Capability. The following pattern is required: SysC<*capability number*>: <*capability name*> <*capability number*> is a unique integer number. <*capability name*> begin with an action verb which has optionally an object.
Summary*		Detailed description of the Capability. Can be expressed as a user story. For example: As <*actor name*>, the system helps me to <*system capability*> in order to carry out <*mission*>. As <*actor name*>, I want to <*receive benefit*> from the system, when <*precondition*>, so that I can <*mission*>.
Mission	(M)	Mission(s) related to the Capability as defined in the chapter 4.2.
Involved actors*		It includes human, non-human actors, and the system itself. Actors are defined in the chapter 4.1. Minimum one actor is referenced.
ERTMS Level	(M)	Applicable ERTMS Level: 0, 1, 2, 3 or NTC.
GoA	(M)	Applicable level of automation: 1, 2, 3 or 4.
Pre-condition*		Conditions necessary for the Capability to be performed. Must be selected from the list of Constraints and/or States in the System Analysis. It can be expressed as: • a constraint that evaluates to TRUE or, • an entering state of the system. A constraint is a Capella element which allow to formalize list of conditions carried by an element (function, a functional exchange). E.g.: 
Condition during execution		
Post-Condition*		Conditions verified after the Capability has been performed. Must be selected from the list of Constraints and/or States in the System Analysis. It can be expressed as: • a constraint that evaluates to TRUE or,

		• an existing state of the system. It is a measurable or observable result delivered by the system to an actor.
Non-Functional Requirement		Non-functional requirement applicable to the capability.
Functional Chain*	(M)	Functional chain(s) describing the Capability. Functional chain is an ordered sequence of functions linked with directional functional exchange or sequence link. Minimum one functional chain referenced. Generally, one functional chain by system mode where the capability is operating.
Scenario	(M)	Scenario describes a time sequence of exchange between actors (exchange scenario) or functions (functional scenario).
Available in Modes and States		The list of Modes and States of the system in which the current System Capability is active.
Operational Phase	(M)	Reference to an operational phase, as identified in chapter 3.

4.3.2 Example

4.3.2.1 System Capability Group 1: Train Protection

4.3.2.1.1 System Capability SysC01: Supervise train speed and movement without train integrity guarantee

Name*	(M)	SysC01: Supervise train speed and movement without train integrity guarantee.
Summary*		As a train, the system supervises my speed and movement in order to allow me to move safely.
Mission	(M)	Control safe train movement.
Involved actors*		+Movement authority Transactor +WSA (EB) +TRain DRiver +Physical Train Unit
ERTMS Level	(M)	0, 1, 2, 3, NTC.
GoA	(M)	1, 2, 3, 4.
Pre-condition*		MT has received information from the interlocking. Enter in a Functional ETCS mode (SR, OS, LS, FS, ..., or in a Technical mode (SB, SL, PS, …)
Condition during execution		
Post-Condition*		End of mission. EB triggered.
Non-Functional Requirement		Safety level: SIL4 Delay between passing an EOA/LOA and applying the emergency brake (subset 041 §5.2.1.13) <1sec.
Functional Chain*	(M)	[SFCD] Supervise train speed and movement without train integrity guarantee [SAB] Supervise train speed and movement without train integrity guarantee
Scenario	(M)	[SES] Supervise train speed and movement without train integrity guarantee
Available in Modes and States		Mode ETCS FS/OS/SH/SB/LS/SR
Operational Phase	(M)	Movement between station