Ref. Ares(2026)2526093 - 04/05/2026

Rail to Digital automated up to autonomous train operation

D10.3 – ATO GoA3/4 system test specification and report

Due date of deliverable: 30/11/2025

Actual submission date: 27/03/2026

Leader/Responsible of this Deliverable: Michele Bruzzo (HITACHI RAIL STS)

Reviewed: **Y**/N

Document status		
Revision	Date	Description
01	02/10/2025	First issue
02	28/11/2025	Second issue containing validation outcomes of prototypes
03	20/02/2026	Partners' contribution collection
04	27/03/2026	Update following 2nd internal review cycle outcomes. Release for TMT review.

Project funded from the European Union's Horizon Europe research and innovation programme		
Dissemination Level		
PU	Public	X
~~**SEN**~~	~~Sensitiv – limited under the conditions of the Grant Agreement~~	

Start date: 01/12/2022 Duration: 42 months

Interne

ACKNOWLEDGEMENTS

This project has received funding from the Europe's Rail Joint Undertaking (ERJU) under the Grant Agreement no. 101102001. The JU receives support from the European Union's Horizon Europe research and innovation programme and the Europe's Rail JU members other than the Union.

REPORT CONTRIBUTORS

Name	Company	Details of Contribution
Michele Bruzzo	STS	Task 10.3 Leader and author of contributions in chapter 2.6, as well as review of other contributions
Monica Bordo	STS	Author of contributions in chapter 4, as well as review of other contributions
Julien Plantard	ATSA	Author of contributions in chapter 2.1, as well as review of other contributions
Pierre-françois Jauquet	ATSA	Author of contributions in chapter 2.1, as well as review of other contributions
William Busuttil	ATSA	Author of contributions in chapter 2.1, as well as review of other contributions
Petr Stříteský	AZD	Author of contributions in chapter 2.2, as well as review of other contributions
Xabier Larrea Jauregui	CAF	Author of contributions in chapter 2.3 as well as review of other contributions
Serra Diana	FS	Author of contributions in chapter 2.4, as well as review of other contributions
Pierre Le Maguet	GTSD	Author of contributions in chapter 2.5, as well as review of other contributions
Jad Haber	SMO	Author of contributions in chapter 2.7, as well as review of other contributions
Angel Fernandez Gago	ADIF	Author of Executive Summary, Introduction. Reviewer
Francisco Haro Fortes	RENFE	Author of Executive Summary, Introduction. Reviewer
Diego García Vaquero	RENFE	Author of Executive Summary, Introduction. Reviewer
Jeroen Braakman	NS	Reviewer
Lucia Maria Cozzolino	FS	Reviewer
Salvatore Barretta	FS	Reviewer
Lorenzo Barruffo	FS	Reviewer

Interne

Disclaimer

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The content of this document reflects only the author’s view – the Joint Undertaking is not responsible for any use that may be made of the information it contains. The users use the information at their sole risk and liability.

Interne

EXECUTIVE SUMMARY

Context and Objectives

The transition toward fully automated rail operations is a strategic priority for the European rail sector, aiming to improve safety, efficiency, and interoperability across national systems. Within this context, Task 10.3 of the R2DATO project addresses the validation of ATO (Automatic Train Operation) technologies up to Grade of Automation 4 (GoA4). The task is part of Work Package WP10, which focuses on prototype development, and specifically aims to ensure that the developed ATO GoA3/4 systems meet the functional and architectural specifications defined in WP6.

The deliverable D10.3 provides a structured framework for validating these prototypes, ensuring that they are technically sound, interoperable, and ready for integration into future demonstrators. This validation is essential to mitigate risks before deployment and to support the broader EU goals of digitalizing and automating rail transport.

It is important to note that the validation activities in WP10 are distinct from the work carried out in WP35. While WP10 focuses on prototype-level functional validation of ATO GoA3/4 systems against WP6 specifications, WP35 addresses broader topics related to test bench development, system-level validation methodologies, and certification processes. The results of Task 10.3 therefore complement, rather than overlap with, the activities in WP35.

Scientific/Technical Approach or Methodology

Task 10.3 is divided into two core sub-tasks:

1. Definition of System Test Specifications: this sub-task translates updated functional specifications into operational scenarios and test sequences. These are designed to reflect realistic use cases and ensure comprehensive coverage of system functionalities.

   As part of this activity, al most all partners have defined a multi-layer validation approach combining Software-in-the-Loop (SIL), Hardware-in-the-Loop (HIL), ensuring verification of the prototype's technical readiness required for the subsequent testing phase in a real test environment, where applicable.

2. Integration and Execution of System Tests: the defined test sequences are executed in a controlled laboratory environment using the ATO GoA3/4 Reference Test Bench, which emulates system interfaces and monitors outputs. The test bench enables consistent and repeatable validation across different prototypes.

The task is led by HITACHI, with contributions from CAF, ADIF, AZD, GTSD, FS, SMO, MERMEC, FT, ATSA, and NSR. Each partner is responsible for providing test activity descriptions, methodologies, and feedback, while reviewers ensure coherence and quality of the final deliverable.

Main Conclusions

The validation activities under Task 10.3 have successfully demonstrated the core functionalities of the ATO GoA3/4 system solution. Key results include:

- Operational readiness of prototypes under defined test scenarios.
- Evidence of compliance with the specifications in terms of functions and interfaces, forming the basis for interoperability across prototypes developed by different partners.
- Robustness of the testing framework, which supports modular and scalable validation.

Interne

- Successful validation of core functionalities such as ADM, APM, REP, as well as the communication interfaces between these systems, ensuring that ATO solutions meet the functional requirements.
- Consolidation of feedback for specification refinement, resulting in concrete proposals for improving interface definitions, packet structures, and operational sequences.

The deliverable includes a dedicated chapter consolidating the joint common findings identified across partners, which are relevant for future amendments of the ATO GoA3/4 specifications.

The deliverable consolidates high-level test specifications, execution summaries, and results, offering a comprehensive overview of the validation process while maintaining confidentiality of proprietary data.

In addition, these validation activities directly support the demonstrator work packages (WP38, WP39, and WP46), ensuring faster integration and improved readiness for field testing through the use of validated prototypes.

Recommendations and Indications for the Future

While Task 10.3 has met its validation objectives, further research and innovation (R&I) is recommended to:

- Extend testing to full-scale demonstrators in operational environments.
- Support industrialization and market deployment of ATO GoA3/4 technologies.
- Contribute to standardization and regulatory alignment at the European level.
- Enhance market acceptance through continued validation and stakeholder engagement.
- Strengthen test environments as soon as more mature specifications and interfaces become available.

The work completed in Task 10.3 provides a solid foundation for future development and deployment of automated rail systems, reinforcing the EU's leadership in rail innovation and digital transformation.

ABBREVIATIONS AND ACRONYMS

ADM	Automatic Driving Module
APC	Automation Processes Cluster
APM	Automatic Processing Module
ATO	Automatic Train Operation
ATP	Automatic Train Protection
AV	Available
CCS	Command Control and Signalling
EDITA	Experimental Driverless Vehicle for Innovative Technologies of AZD
ERJU	Europe's Rail Joint Undertaking
ERTMS	European Rail Traffic Management System
ETCS	European Train Control System
EVC	European Vital Computer
FA State	Failure State
FIS	Functional Interface Specification
FS mode	Full Supervision mode
FS	Ferrovie dello Stato
FSM	Finite State Machine
GoA	Grade of Automation
GUI	Graphical User Interface
JP	Journey Profile
LX	Level Crossing
LZ	Localization
MP	Mission Profile
OAS	On-board Automation System
OS mode	On Sight
PAL	Protection Adaptation Layer
PER	PERception
R2DATO	Rail to Digital automated up to autonomous train operation
RAP	Roll Away Protection
REP	Repository
SP	Segment Profile

Interne

SS	SubSet
TCG	Task related to Change of GoA
TCMS	Train Control Management System
TCP	Task related to CouPling
TDO	Train Door Operation
TJP	Task related to Journey Profile
TRL	Technical Readiness Level
TSI	Technical Specification for Interoperability
TST	Task related to Setting Train context state
TUC	Task related to UnCoupling
WP	Work Package

TABLE OF CONTENTS

Acknowledgements 2
Report Contributors 2
Executive summary 4
Abbreviations and Acronyms 6
Table of Contents 7
List of Figures 8
List of Tables 10
1 Introduction 11
2 Description of Testing and Validation activities 13
2.1 ATSA 13
2.1.1 Methodology 13
2.1.2 Test Activities Description 14
2.1.3 Feedback for Specification and/or Demonstrators 14
2.2 AZD 16
2.2.1 Methodology 16
2.2.2 Test Activities Description 24
2.2.3 Feedback for Specification and/or Demonstrators 39
2.3 CAF 46
2.3.1 Methodology 46
2.3.2 Test Activities Description 47
2.3.3 Feedback for Specification and/or Demonstrators 68
2.4 FS 70
2.4.1 Methodology 70

Interne

2.4.2 Test Activities Description 71
2.4.3 Feedback for Specification and/or Demonstrators 83
2.5 GTSD 86
2.5.1 Methodology 86
2.5.2 Test Activities Description 88
2.5.3 Feedback for Specification and/or Demonstrators 92
2.6 HITACHI 96
2.6.1 Methodology 96
2.6.2 Test Activities Description 98
2.6.3 Feedback for Specification and/or Demonstrators 109
2.7 SMO 114
2.7.1 Methodology 114
2.7.2 Test Activities Description 114
2.7.3 Feedback for Specification and/or Demonstrators 116
3 Conclusions 117
4 Cross-Partner Findings and Proposed Amendments to the ATO GoA3/4 Specifications 119
5 References 121

LIST OF FIGURES

Figure 1 : ATSA FP2-R2DATO – Development of the ATO GoA4 System prototype & Test Platform 13
Figure 2: AZD example component structure 16
Figure 3: AZD TCMS emulator “MERKUR” (device + DMI) 17
Figure 4: AZD TCMS emulator's IO modules 18
Figure 5: AZD TCMS Emulator UI (one of many pages) 19
Figure 6: AZD ETCS-OB simulation 20
Figure 7: AZD ATO-TS simulation 21
Figure 8: AZD “EDITA” - Experimental Driverless Vehicle for Innovative Technologies of AZD 21
Figure 9: AZD "EDITA" towing a driverless train with passengers 22
Figure 10: AZD visualization of "EDITA" interior 22
Figure 11: AZD "Kopidlnka" line - stops and stations 23
Figure 12: AZD "Kopidlnka" line - altitude diagram 23
Figure 13: AZD Joint computer for GoA3-4 prototypes - design 24
Figure 14: AZD Joint GoA3-4 computer - train installation 25
Figure 15: AZD ADM/ATO unit 26
Figure 16: AZD R1 rack containing TCMS, ADM, anti-wheelslip, PAL and REP 27

Interne

Figure 17: AZD PAL User Interface (unfinished) 29
Figure 18: AZD Integration of Perception into the vehicle 31
Figure 19: AZD Visualized output of PERception sensoric evaluation of area in front of the train .. 31
Figure 20: AZD A modified final system architecture used as a reference for testing environment 33
Figure 21: AZD Lab in imagination 34
Figure 22: AZD Lab in reality 34
Figure 23: CAF methodology diagram 47
Figure 24: Target environment HW setup 48
Figure 25: Developed prototypes and interfaces 50
Figure 26: APM validation architecture 51
Figure 27: ADM validation architecture 52
Figure 28: CAF APM test results 66
Figure 29: CAF ADM test results 68
Figure 30: FS V Cycle Model 70
Figure 31: FS Focus on Automatic Driving Module (red box) 71
Figure 32: FS Test Environment 72
Figure 33: FS GUI for plots and commands 73
Figure 34: FS Track speed limits 74
Figure 35: FS Gradient and curvature for the line 74
Figure 36: FS UC ID 1 sequence diagram 77
Figure 37: FS UC ID 2 sequence diagram 78
Figure 38: FS UC ID 3 sequence diagram 79
Figure 39: FS UC ID 4 sequence diagram 80
Figure 40: FS UC ID 5 sequence diagram 81
Figure 41: GTSD process 88
Figure 42: GTSD standard test environment for vehicle products 89
Figure 43: GTSD WP10-WP11 tayloring of GTSD's vehicle test environment 90
Figure 44: APM and PER test 91
Figure 45: HITACHI process 97
Figure 46: HITACHI Relation between the System and the Equipment under test (with red border) 99
Figure 47: HITACHI Target environment GUI 100
Figure 48: HITACHI Laboratory architectures for ADM and REP validation 101
Figure 49: HITACHI Emulation process within SimATO 102
Figure 50: HITACHI ADM test results 107
Figure 51: HITACHI REP test results 109
Figure 52: SMO 19-inch rack for testing the ATO-OB (left) and the AVVO-OB (right) 115

Interne

LIST OF TABLES

Table 1: CAF APM validation tests 65
Table 2: CAF ADM validation tests 68
Table 3: CAF Errors and feedback detected in the specification 69
Table 4: FS Basic Passenger JP Data 75
Table 5: FS Basic Freight JP Data 75
Table 6: FS Use Cases and Functions Mapping 82
Table 7: FS Summary of Status of Use Cases Test Planning and Execution Activity 82
Table 8: FS Summary of Status of Test Planning and Execution Activity 83
Table 9: FS Feedback for R2DATO WP6 Specifications 84
Table 10: FS Change Requests for SS125 and SS130 85
Table 11: Improvement suggestions by GTSD 95
Table 12: HITACHI List of test defined for ADM with reference to requirements of “ATO up to GoA4 SSRS” 104
Table 13: HITACHI List of test defined for ADM with reference to requirements of SS125 106
Table 14: HITACHI List of test defined for REP with reference to requirements of “ATO up to GoA4 SSRS” 109
Table 15: HITACHI List of improvements suggested to stabilize REP communication principles 110
Table 16: HITACHI List of improvements suggested to stabilize ADM and REP requirements 111
Table 17: HITACHI List of improvements suggested to stabilize ADM and REP interfaces 113

Interne

1 INTRODUCTION

The present document constitutes Deliverable D10.3 – ATO GoA3/4 System Test Specification and Report, developed within the framework of Task 10.3 of Work Package WP10 under the Flagship Project FP2 – R2DATO. WP10 contributes to the Technical Enabler for Automated Driving (ATO Technologies), supporting the broader objectives of the Europe's Rail Joint Undertaking (ERJU) programme. This deliverable also aligns with the system architecture defined in WP6, particularly the specification document D6.5, and supports the validation of ATO functionalities up to Technology Readiness Level (TRL) 6.

Task 10.3 focuses on the validation of ATO GoA3/4 prototypes, which were developed and upgraded in Task 10.1. The aim is to demonstrate the system's capabilities, interoperability, and interchangeability through structured testing activities. These validations are essential to ensure that the ATO solutions meet the functional requirements and are ready for integration into future demonstrators, thereby reducing deployment risks and supporting the EU's strategic goals for rail automation.

A key reference for these validation activities is the System Requirements Specification (SRS) [10], which defines the detailed functions, inputs and outputs, and system requirements used across the APM, ADM, and other subsystems. The SRS provides the authoritative baseline ensuring consistency, completeness, and traceability throughout the testing process.

The motivation for this deliverable stems from the need to bridge the gap between prototype development and operational deployment. While previous work in WP6 defined the functional specifications and architecture, and Task 10.1 focused on building and upgrading the prototypes, Task 10.3 ensures that these systems are rigorously tested in a controlled environment using the ATO GoA3/4 Reference Test Bench. This approach allows for repeatable, consistent validation and provides actionable feedback for further refinement.

A dedicated chapter has also been included to consolidate the joint common findings emerging across partners, providing inputs relevant for future specification amendments.

The structure of this document is as follows:

- Chapter 1: Introduction to the task.
- Chapter 2: Description of the testing and validation activities carried out by each contributor, detailing the methodology, test execution, and feedback collected during the process.
- Chapter 3: Conclusions regarding the activities described in Chapter 2.
- Chapter 4: Joint common findings relevant for future specification amendments.
- Chapter 5: List of referenced documents cited throughout the deliverable.

This deliverable introduces several novel aspects:

- A harmonized validation framework across multiple partners and platforms.
- Modular test sequences tailored to operational scenarios.
- A collaborative approach to feedback and specification refinement.

While the validation activities have confirmed the core functionalities of the ATO GoA3/4 system, some limitations remain, particularly in extending testing to full demonstrator environments. These

Interne

open challenges are acknowledged and will inform future research and innovation efforts, including industrialization, standardization, and market deployment.

2 DESCRIPTION OF TESTING AND VALIDATION ACTIVITIES

2.1 ATSA

2.1.1 Methodology

For the R2DATO WP46 Regional Demonstrator, ALSTOM utilises an existing ALSTOM ETCS application, along with its associated ETCS tools, to simulate ATP supervision, train behaviour, and track conditions. In addition to these existing components, new applications have been developed to implement trackside tools, such as Digital and train transactors, to simulate data exchange between ATO GoA4 Onboard and Trackside (Mission Profiles, Train Data, Segment Profiles ...).

The development of the ALSTOM Main Line ATO GoA4 solution is based on the architecture defined in Deliverable D6.5 and complies with the System Requirements Specification (SRS) version 1.1.0. As part of this development, several key modules have been implemented: the Automated Processing Module (APM), the Repository (REP), the Protective Abstractive Layer (PAL), and the Automatic Driving Module (ADM). The ADM corresponds to the ATO-OB GoA2 functionality as specified in SS125 V1.0.0.

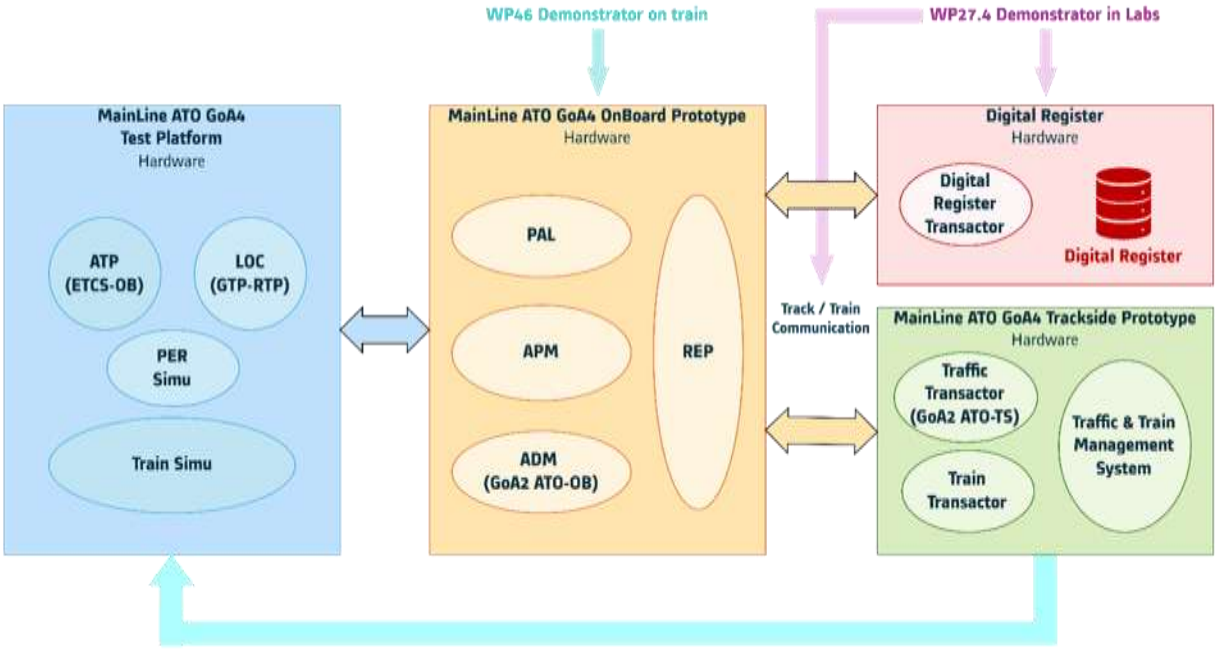

Figure 1 : ATSA FP2-R2DATO – Development of the ATO GoA4 System prototype & Test Platform

Figure 1 shows the architecture of Alstom's GoA4 system prototype and test platform. Validation of the ALSTOM Main Line ATO GoA4 solution takes place on a dedicated simulation platform that precisely replicates the operational environment. This includes integration with systems such as the Train Control and Monitoring System (TCMS), PERception system (PER), the Advanced Safe Train Positioning (ASTP), and the European Train Control System (ETCS). This simulation platform allows comprehensive testing of the ATO GoA4 modules, ensuring full alignment with the objectives and scope defined in WP46.

Interne

2.1.2 Test Activities Description

The test plan is structured to validate the system in sequential steps, starting with simulated environment in laboratory toward deployment on train (the associated Test Report [1] documenting all laboratory V&V activities is available for audit purposes only).

Initially, each module under test is validated as a black box on ATSA's simulated platform. These tests focus on verifying SRS functionalities to ensure WP46 and WP27.4 use cases/Test scenarios coverage and interchangeability. Remote integration tests have been conducted with AZD (WP46) to validate the ability of AZD and ATSA modules to communicate and exchange data. This important and crucial stage in securing the on-train test (WP46) and the demonstration of the interchangeability of most of the ATO GoA4 Onboard modules enabled ATSA to identify some discrepancies in the implementation of these interfaces compared with the interface specifications defined together on the basis of SRS 1.1.0 and sometimes giving rise to the need to clarify and/or change a few specification points.

Following successful validation in a simulated environment, remote integration tests were conducted with R2DATO partners to validate the communication and data exchange between modules. This phase is critical for ensuring seamless interoperability across under tests modules on real environment.

The next steps involve:

- For WP46, on-site testing, where the equipment is installed on the AZD EDITA railbus to execute the defined use cases/Test scenarios in a real operational environment
- For WP27.4, laboratory testing where the Digital Register including the DR transactor developed by DB will replace the one developed by ATSA to execute the defined use cases/Test scenarios in a mixed configuration.

Testing in Lab of the overall ATO GoA4 System prototype has been almost fully tested as of December 2025 and was fully tested based on the test scenarios with the target WP46 Mission-, Journey- & Segment Profiles by the end of February 2026.

WP46 on-train testing is planned over a period extending from mid-January 2026 to early March 2026 while WP27.4 laboratory testing is planned on week 14 – 2026 (the week starting 30 March).

The overall test strategy followed ensures a robust and phased validation of the ATO GoA4 modules under tests, minimizing risks and maximizing reliability. While there was some work to be done to cover gaps in the SRS V1.1.0 (as it relates to the use cases tested), this strategy allowed Alstom to verify and validate all use cases in the lab before going on-train (for WP46), and to ensure that the relevant interfaces with AZD's modules (WP46) worked at a level that allowed interchangeability of the onboard modules.

2.1.3 Feedback for Specification and/or Demonstrators

Alstom believes that the development of future demonstrators needs to be based on clear, precise and complete specifications. Any future R2DATO project schedule must be compatible with this requirement. When demonstrator development occurs concurrently with specification development, (which was the case with the FP2-R2DATO WP46 demonstrator), it creates a need for the contributors to perform heavy and costly activities (interface specifications, sequence diagrams, etc.) with no guarantee of reusability.

Interne

At the European level, there is a need to specify standard working procedures for train drivers (which would be automated in the GoA3/4 SRS) to guarantee that the mainline GoA3/4 solutions developed are interoperable across the European railway network. At present, train drivers are required to follow the traffic regulations set by the (national) Infrastructure Manager for the country they are driving in. While the deployment of ETCS standardizes signalling-related procedures, there are many non-signalling-related tasks whose rules vary from country to country. For example, in Belgium, a train attendant is permitted to close the train doors to depart a station as soon as the planned departure time (in the timetable) has been reached. In contrast, the procedure in the Netherlands requires the driver/train attendant to also have received an MA to depart the station before starting the door closing procedure. If this procedure were not standardized at the European level, it would create a need for suppliers to create country-specific ATO modules, and to dual-fit these modules on trains operating in multiple countries.

ATSA also sees a need to specify the conditions for using the ADM in ATO GoA4 in degraded scenarios. Many degraded procedures require the driver to operate in an ETCS mode other than FS (e.g. SR, OS, or SH). Examples include the need to engage the ATO in SR mode before passing a first balise, or the need to drive in OS mode on IM authorization in the event of a Track Train Detection Device failure. In some degraded scenarios where the train remains in FS mode (e.g. passing an unprotected level crossing close to a stopping point in Journey Profile), it is sometimes not possible to receive a sufficiently long MA to engage the ADM to depart the planned stopping point in the JP (using an ADM complying with TSI-2023).

In addition, such lack in ATO GoA4 specification offers the freedom to let a driver carry out certain operations on a GoA4 ATO demonstrator where, by default, there should be no driver. This obliges the contributors to demonstrators to make specific developments which may constitute significant operational regressions compared with more advanced options/technical solutions already validated on other internal or external research projects.

At the APM level, the process for determining the running direction in the SRS V1.1.0 is not fully reliable, and sometimes sets up the wrong cabin for the TJP to be executed. Under the SRS, the APM has information about the orientation in which the LRBG was passed (nominal or reverse, with variable Q_DIRLRBG), and compares that to the direction that that LRBG is supposed to be passed when executing the TJP. If these directions match, then the APM is supposed to open the last cab reported active (otherwise it opens the other cab). However, this process assumes that the last active cab was the one that was open at the time the LRBG was passed. If this is not the case, then the APM will open the wrong cab on the consist. This can create issues with the reliability of this procedure at the beginning of a mission, where a cab may have been opened manually (without being associated with a TJP in the previous mission).

Interne

2.2 AZD

2.2.1 Methodology

2.2.1.1 Self-Testing

The self-testing process involves an “internal” testing of specific functions, modules and components, and does not rely on any external environment. All inputs and outputs on various levels can be generated internally by the component under test. A specific test scenario is created within the component itself, emulating a specific input, while having strictly defined what the output should be by implementing a specific method for evaluation. The inputs may be represented by simulated events or data reception, while not only the result, but also the efficiency (e.g. processor time required to process the data and generate response) is measured. On the picture below (Figure 1), an example schema of two tested components with shared resources and their internal structure is shown. The testing scenarios are applied gradually from lowest to highest levels - from testing each individual method, through testing one module at a time, to connecting already self-tested modules within one component against each other and testing the whole component with all modules included. Fictional example of self-test evaluation: Serialization of data packet completed under the set limit of 10ms with buffer length of 23 bytes, where the buffer must be 0xabcdefabcdef….and the CRC computed must be 0xDEF8432D.

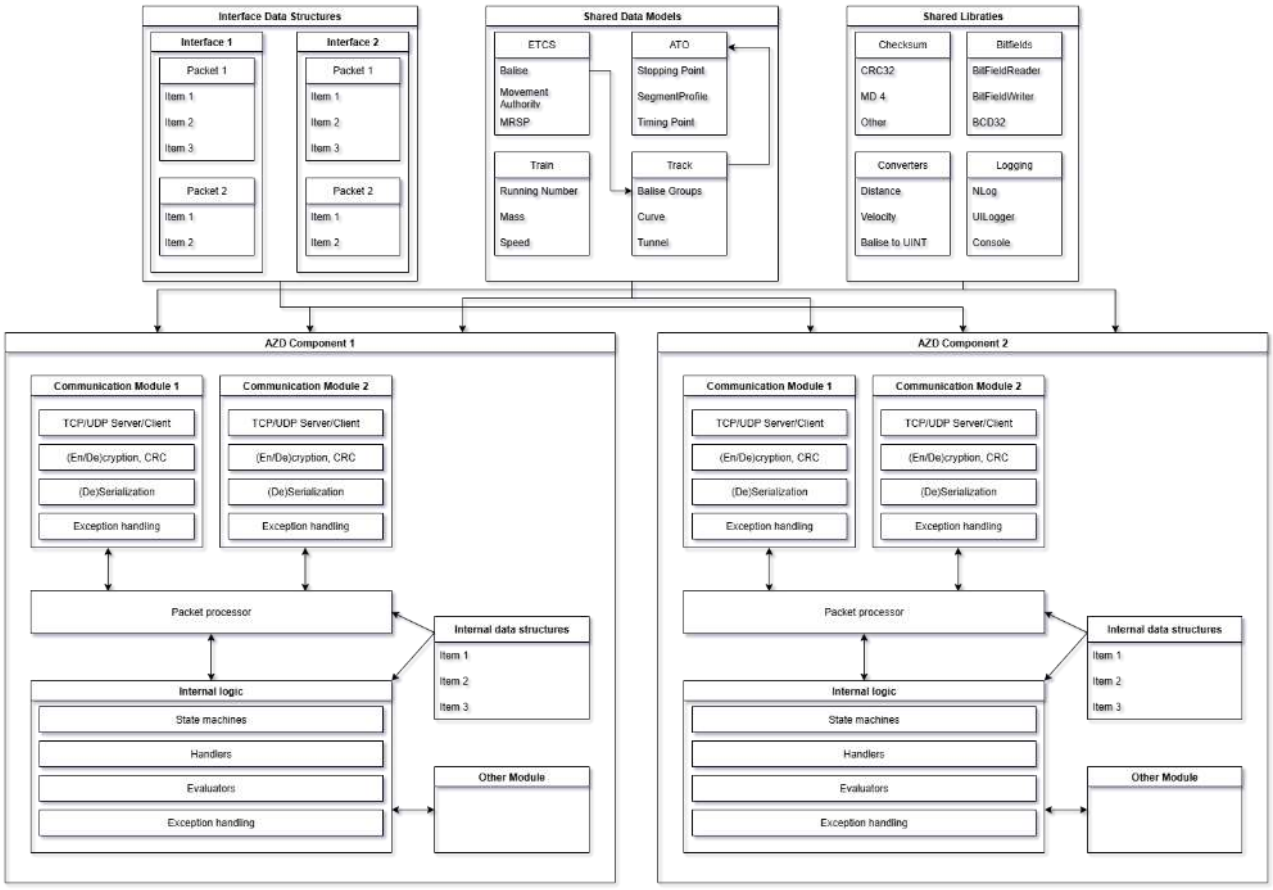

Figure 2: AZD example component structure

2.2.1.2 Laboratory Testing

After each component is validated internally, it is subsequently tested in the laboratory against other components of the system, which may be either simulated or real industrial devices. The choice between simulated or real devices depends on several factors, such as the specific system under test, the physical availability of a given component in the laboratory, the purpose of the test, and the maturity level of the component itself. For instance, the TCMS must be simulated in the laboratory, as a full train cannot physically fit inside the facility.

The methodology adopted for laboratory testing closely resembles the one used for self-testing: the process starts from simple elements and gradually progresses toward more extensive and complex setups. This approach has consistently proven to be the most effective in identifying the root cause of bugs and other issues. The main difference lies in the evaluation methods. In this phase, defining the testing methodology in the same detailed, input-output-oriented way as for self-tests is not feasible. Instead of focusing on a specific input and expecting a precisely measured output as raw data, the emphasis shifts toward a broader functional perspective. Since the content of the data should already have been validated in earlier phases (although low-level issues may still emerge), the attention moves to verifying the correct sequence of events and checking that all functions are executed properly. A typical observation might be: after receiving the Journey Profile and Segment Profile, the ADM successfully localised upon reading balise group 504:152.

The ultimate goal of laboratory testing is to assemble the entire system, interconnect all of its components, and bring it to a state where it can emulate a complete train run as it would occur on the target train.

Laboratory-specific setup

Here is an overview of components used for testing in the laboratory. If a component is not listed here, it is used both in laboratory and on the vehicle.

TCMS (Emulated)

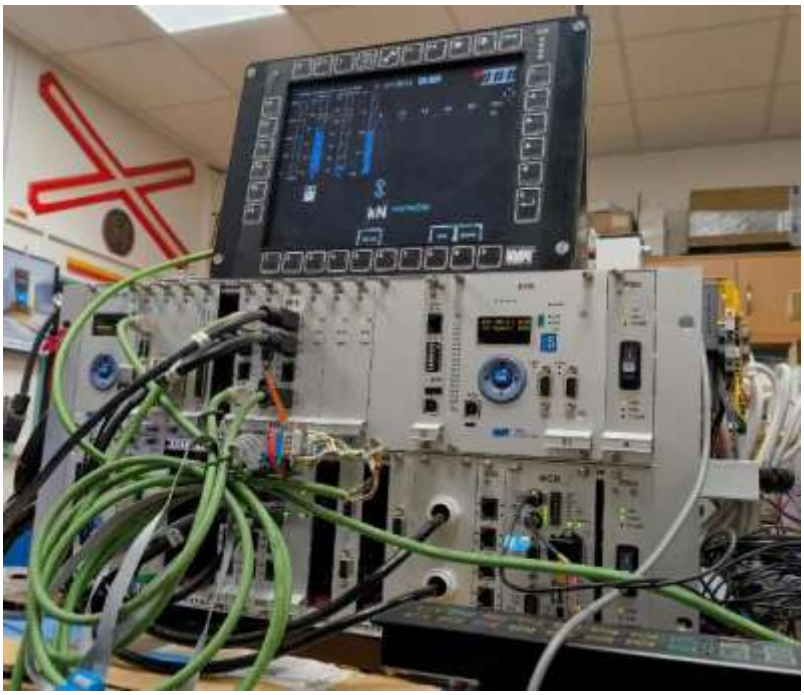

Figure 3: AZD TCMS emulator “MERKUR” (device + DMI)

Interne

This is our latest TCMS emulator, called internally as a “MERKUR” (a reference to a very popular Czech technical construction kit for both children and adults). Compared to previous versions, which needed to be manufactured specifically for given type of vehicle, this TCMS allows to emulate any train – from small diesel railbus to a high-speed electric unit or a heavy freight locomotive. The complete vehicle with all of its specific features, including analog and digital inputs and outputs needs to be defined. Based on that, it is running various simulations – e.g. the whole physical model of both traction and brake, behaviour of engines, doors, lights or even toilets – whatever is managed by TCMS on given vehicle class.

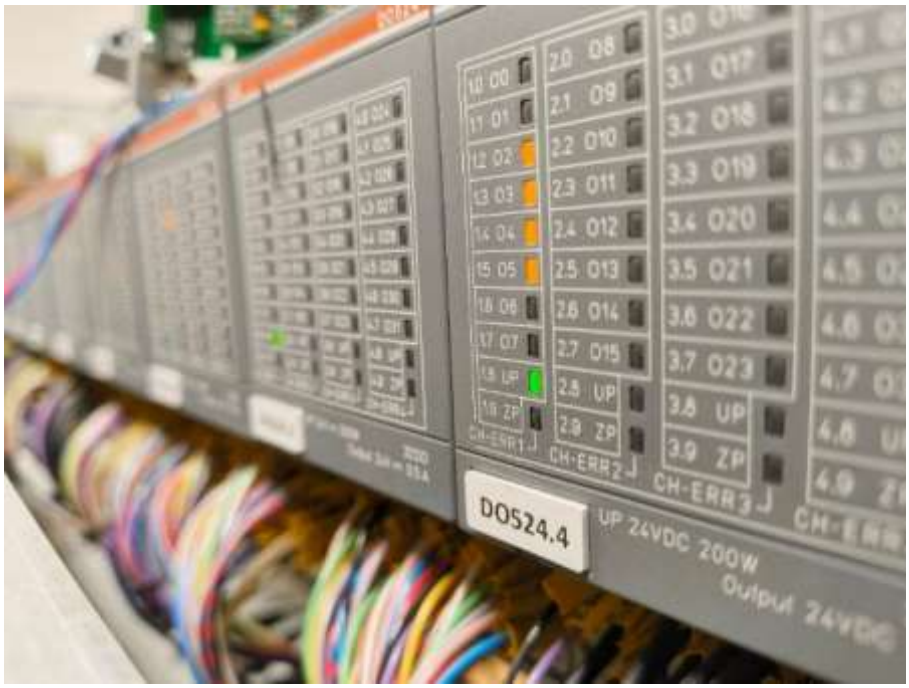

Figure 4: AZD TCMS emulator's IO modules

After that, through a dedicated user interface, it becomes possible to control every controller physically installed on the train. In simple terms, this device emulates the physical train and provides all the elements that a real train would normally deliver, including all CAN and Ethernet interfaces toward the surrounding subsystems. In this way, vehicle-specific DMIs can be connected, and interoperable interfaces can be tested as well.

For the purpose of testing GoA3/4 ATO functionalities, the “MERKUR” system, among other capabilities, implements the ERTMS Subset 139 interface (towards the ADM) and the C55 interface (towards the APM).

Interne

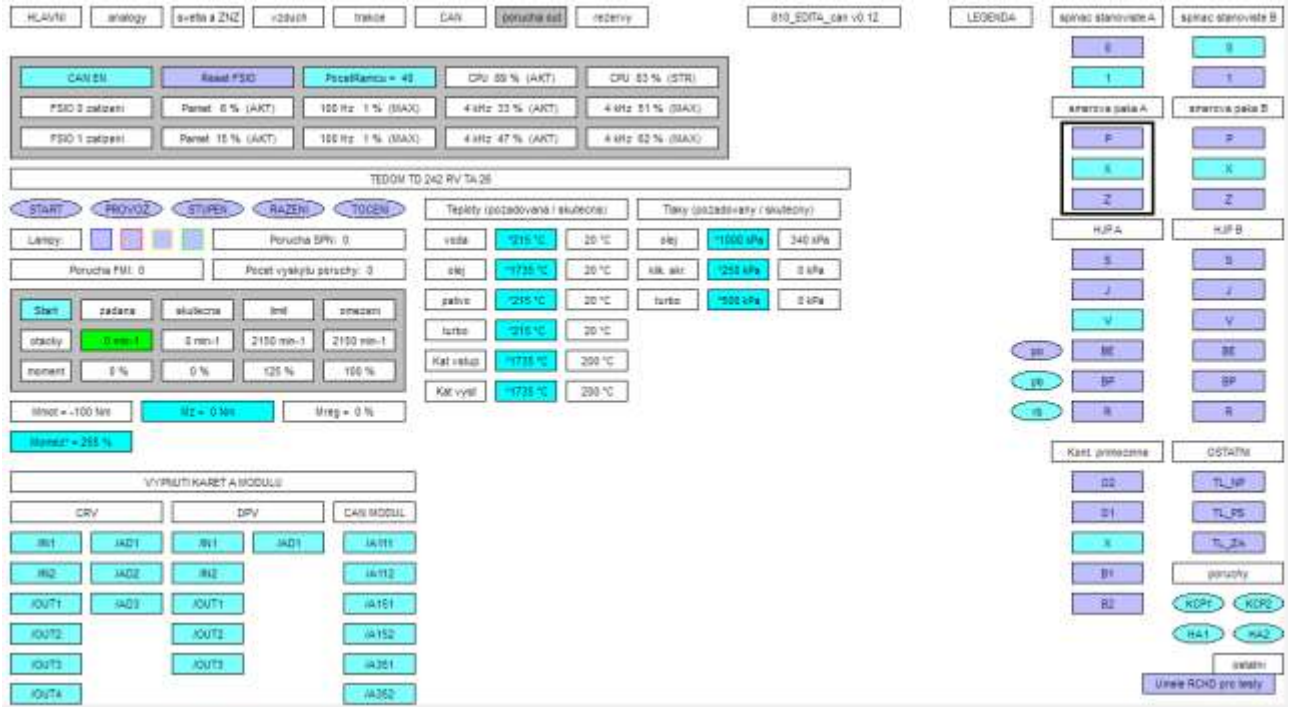

Figure 5: AZD TCMS Emulator UI (one of many pages)

ETCS-OB (Emulated) – ETCS NeXtGen Onboard Simulator

Amongst the components which are present on the train only and therefore needs to be emulated, is the ETCS-OB unit. For this purpose, an ETCS-OB simulator is used. This Windows .NET 5.0 application was originally developed for laboratory testing of ATO and its ERTMS Subset 130 towards ETCS, but eventually, it has served various different purposes. Multiple times, it has been used on board of the train to emulate the ETCS-OB (even if actual device was present). For example, on AZD's test motor coach class 851, it was used to provide Subset 130 to ATO, because the physical ETCS OB mounted on the train did not yet support this interface. It was using some proprietary interface to sniff some information from the real ETCS while using this information to produce SS130 packets. Such use was applicable for testing the ERTMS/ATO on lines equipped with ETCS L2. Later, the simulator has found another purpose – simulating the complete ETCS-OB on lines completely without ETCS equipped. For example, before our test line "Kopidlnka" was equipped with balises, the simulator was reading virtual balises using GPS. Now, when "Kopidlnka" is equipped with ETCS L2 and our test train EDITA implements TSI2023-compatible ETCS-OB with SS130, the ETCS-OB simulator is being used in the laboratories only.

Interne

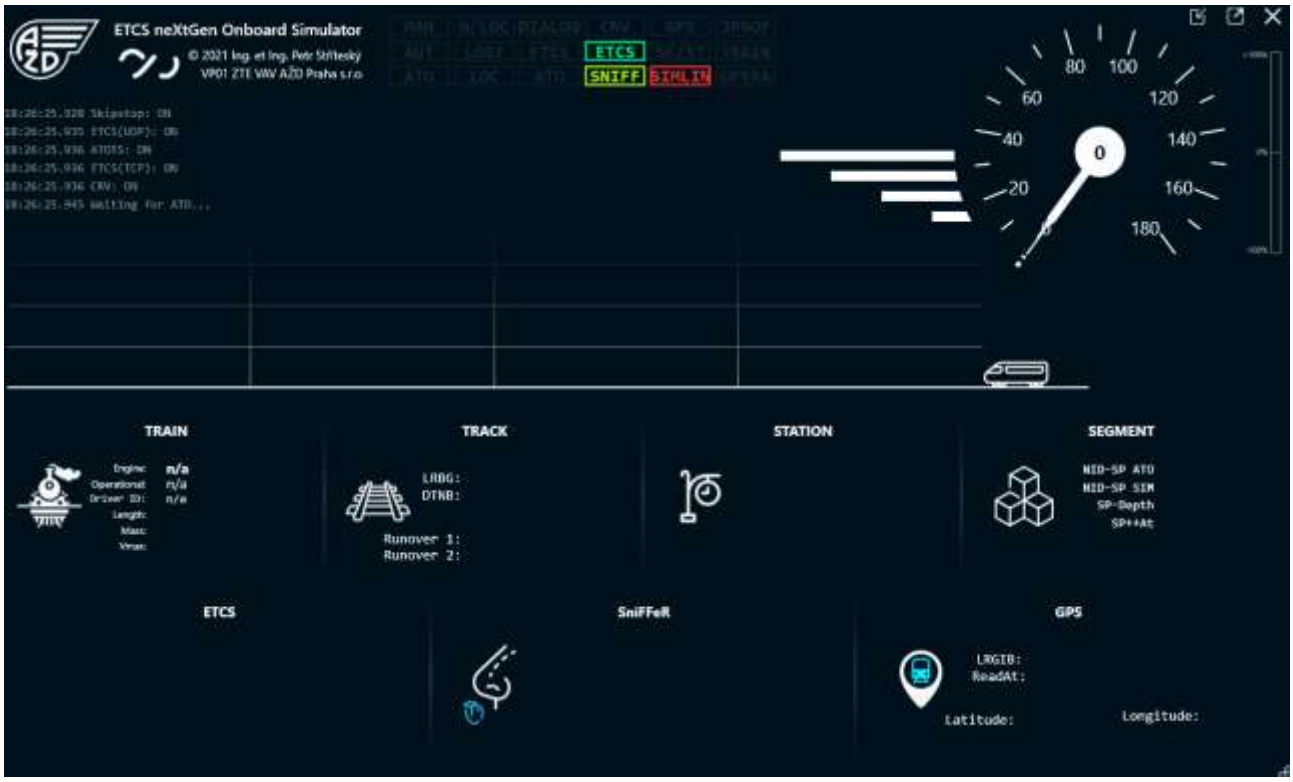

Figure 6: AZD ETCS-OB simulation

ATO-Trackside (Emulated) – AoE NeXtGen TrackFlow

While it is technically possible to connect to an actual ATO-Trackside from the laboratories using a GSM-R router, this option is deliberately not chosen. The real ATO-TS would not provide an emulated train with any operational data. Since the ATO-TS is an operational device, tightly integrated with the TMS, the interlocking, and the dispatcher's route-setting processes, it would not allow researchers to generate non-operational scenarios required for comprehensive testing.

The ATO-TS simulation instead offers significantly greater flexibility, enabling the creation of various situations that would be extremely difficult—or even impossible—to reproduce on an operational railway. Examples include re-using the same Train Running Number multiple times in a single day, or sending exceptionally long Segment Profiles to the ATO/ADM.

The role of the ATO-TS simulator is very similar to that of the ETCS-OB simulator. Although originally designed for laboratory testing of ERTMS Subset 126 and Subset 148, the ETCS-OB simulator has been deployed multiple times on railway vehicles, where it served effectively as an "Onboard Trackside".

Interne

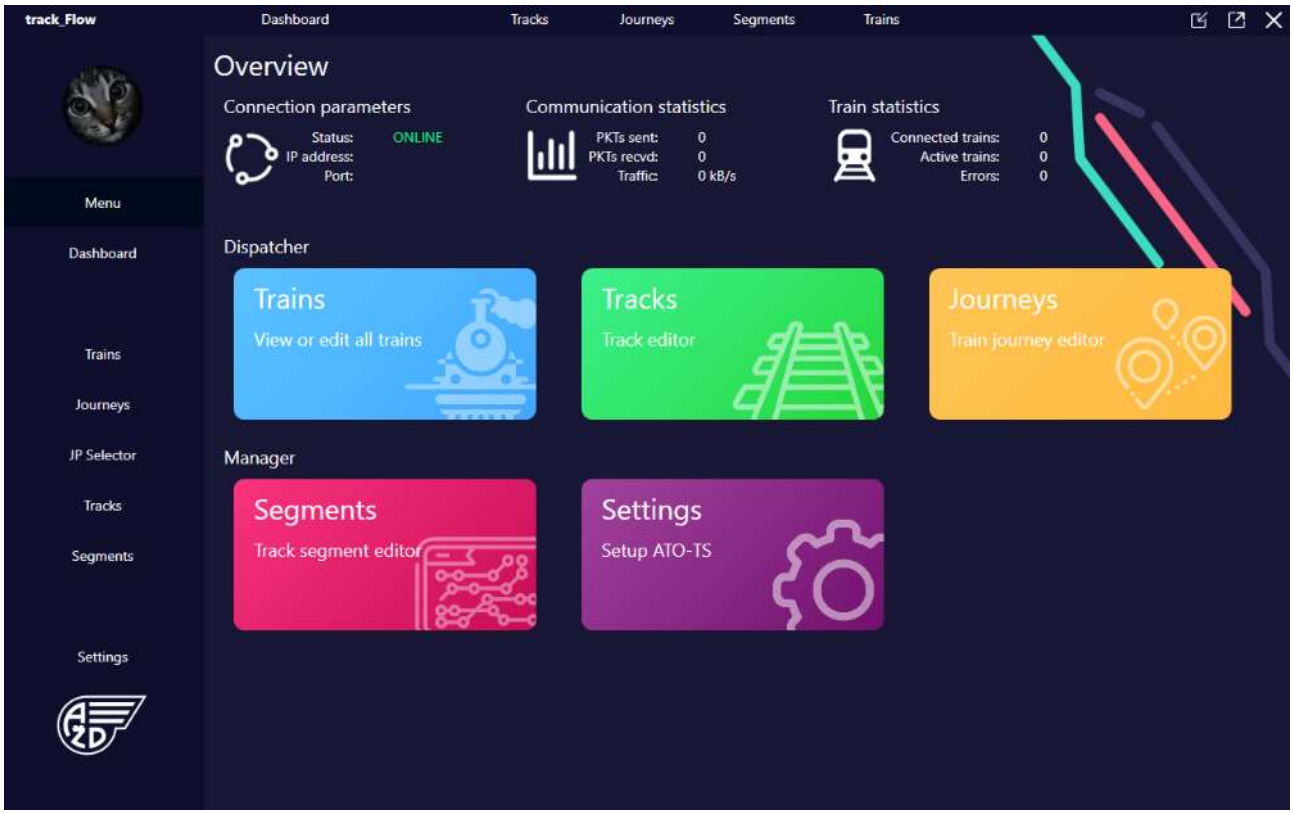

Figure 7: AZD ATO-TS simulation

2.2.1.3 On-Train Testing

After all individual components and the overall system have been validated in the laboratory, the next step is to bring the setup onto an actual train operating on a physical railway line. For the purpose of testing the GoA3-4 specification, in accordance with the newly defined System Requirement Specification for GoA3-4 ATO version 1.1.0 issued by ERJU R2DATO Task 6.5, the test activities are carried out using the “EDITA” test car on the dedicated test line “Kopidlnka”.

Figure 8: AZD “EDITA” - Experimental Driverless Vehicle for Innovative Technologies of AZD

EDITA

"EDITA" stands for Experimental Driverless Vehicle for Innovative Technologies of AZD. It is a completely refurbished 1970's passenger motor coach, turned into a moving railway laboratory for development and testing of GoA 3-4 systems. Except for being used as a test train for research and development, it has been used as a leading vehicle for public autonomous regular passenger service, conducted on the "KopidInka" line in 2025. When necessary, the EDITA can be used also as a standard, manually-driven train. It can be operated by driver in fully manual mode, semi-automatic mode for speed regulation (sort of "cruise control") with driver setting the desired speed, or in automatic mode with GoA2 national system "AVV" or interoperable GoA2 TSI2023-Compliant ERTMS/ATO.

Figure 9: AZD "EDITA" towing a driverless train with passengers

Inside EDITA, there is no room for passengers. It is a moving laboratory for developers and researchers only. It is equipped with all the technology necessary for driverless operation, including the PERception module, TSI2023-compliant ETCS-OBU, more than 10 antennas on the roof for GPS, GSM-R, WiFi, testing of future FRMCS and others… It was remanufactured in a matter to be research and testing-friendly, by being built in a very modular way, so it is easy to (un)install technologies. Inside, there are four racks connected with open cable holders to ease up the installation process.

Figure 10: AZD visualization of "EDITA" interior

KopidInka

Interne

The so called “Kopidlnka”, or a “Kopidlno line”, is a non-electrified single-track regional line located in North-Eastern Bohemia. The line was built in 1883 and is about 23 km long. For most of its existence, the line was owned and operated by the state, until 2010, when the regular service has been discontinued and the line abandoned. Later, in 2014, the line was offered for sale and in 2016 it was sold to a private company AZD Praha. The line is now being converted to line for testing the new railway technologies, especially focusing on automatic train operations. The line starts and ends in stations Kopidlno and Dolní Bousov, which are owned and operated by the national infrastructure manager Správa Železnic. Therefore, the “Kopidlnka” is accessible and is part of the national infrastructure – any train can enter and leave the line. It is a common sight to meet other passenger trains at the endpoint stations.

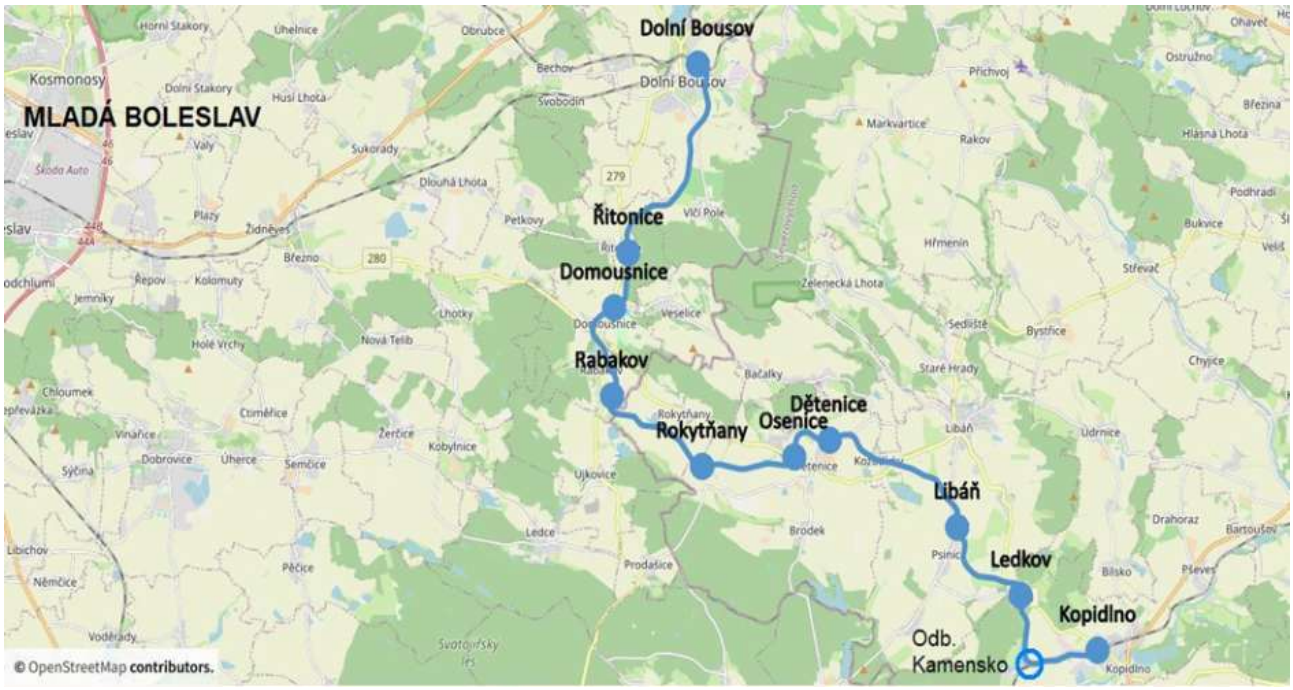

Figure 11: AZD "Kopidlnka" line - stops and stations

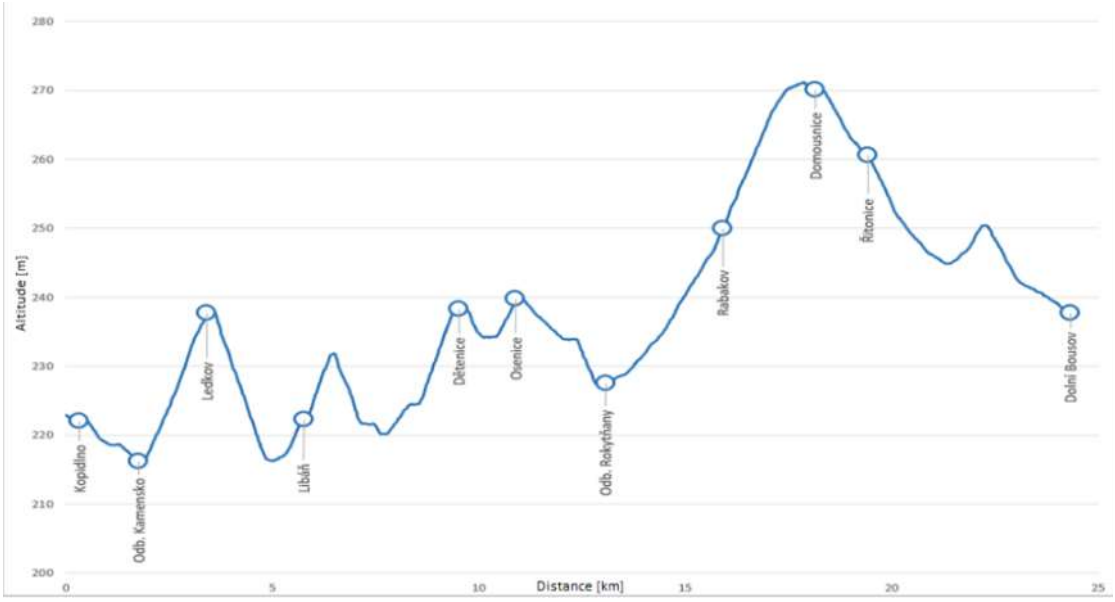

Figure 12: AZD "Kopidlnka" line - altitude diagram

2.2.2 Test Activities Description

2.2.2.1 Equipment Under Test

The equipment to be used consists of prototypes to be tested, along with operational and standardized devices. The setup for external components like TCMS or ETCS may differ in laboratory and on train, but the GoA 3-4 ATO prototypes used in laboratory are the same as used on the train.

Software

For most of the new GoA3-4 prototypes, which are not yet fully standardized, a multi-platform languages and tools were used. That means, the created software can be run on multiple processor architectures, multiple operating systems (or, under some conditions, even without OS). This solution was used in order to be able to run the software on non-railway-dedicated computers for research and development, while expecting to move the software to an industrial computer, after the prototype is finished, standardized, fully validated and certified.

Hardware

Joint computer rack for autonomous systems

For some GoA3-4 prototypes, along with some on-train simulations to help create non-operational scenarios, a following 4-computer rack is used. It consists of 4 RaspberryPi 4B devices with peripherals including an LCD screen, high-speed SSD drives and multi-level ventilation and cooling system. Though it may not be usual to see Raspberry Pi on a railcar, due to the use on multi-platforms tools, it has been proven to be very cost-efficient and surprisingly reliable device to use for testing, until the development is finished. Later, it is expected that we build our own dedicated HW. The Raspberry devices run their factory Raspberry Pi OS on ARM64 architecture. As already mentioned, the developed SW can be run on various devices, including Windows PCs, certified railway machines, but also, for example, a smartphone with Android OS.

The following softwares are mounted on the 4 RPi computers:

- REPository (GoA3-4 prototype)
- Protection Adaptation Layer – PAL (GoA3-4 prototype)
- xOE – Operational Execution emulation (future GoA3-4 trackside component)
- xTP – Train Protection emulation (to be used when ETCS is not available)

Figure 13: AZD Joint computer for GoA3-4 prototypes - design

The rack is manufactured by UCTronics, while it contains modified and extended RPi 4B. It may be powered using 230V, 24V or PoE (Power over Ethernet).

Interne

Figure 14: AZD Joint GoA3-4 computer - train installation

ADM

For the ADM prototype, an industrial railway computer was used. Manufactured by AZD Group's subsidiary company MSV Elektronika, it is a certified computer for use on railway vehicles. In Czechia, it is widely used on railway vehicles equipped with national GoA2 ATO system AVV or interoperable ERTMS/ATO.

APM

For the APM device, by means of hardware, a small portable commercial PC with Windows/Linux is used.

TCMS

A CRV&AVV vehicle control system by AZD's subsidiary MSV Elektronika is used on the test vehicle. It is a modern TCMS providing CAN and ETH interface towards other components, implementing control protocols such as ERTMS/ATO Subset 139 for ADM and SRS 1.1.0 C55 for APM communication.

Figure 15: AZD ADM/ATO unit

Components description

The following chapter describes the components under test from AZD's perspective.

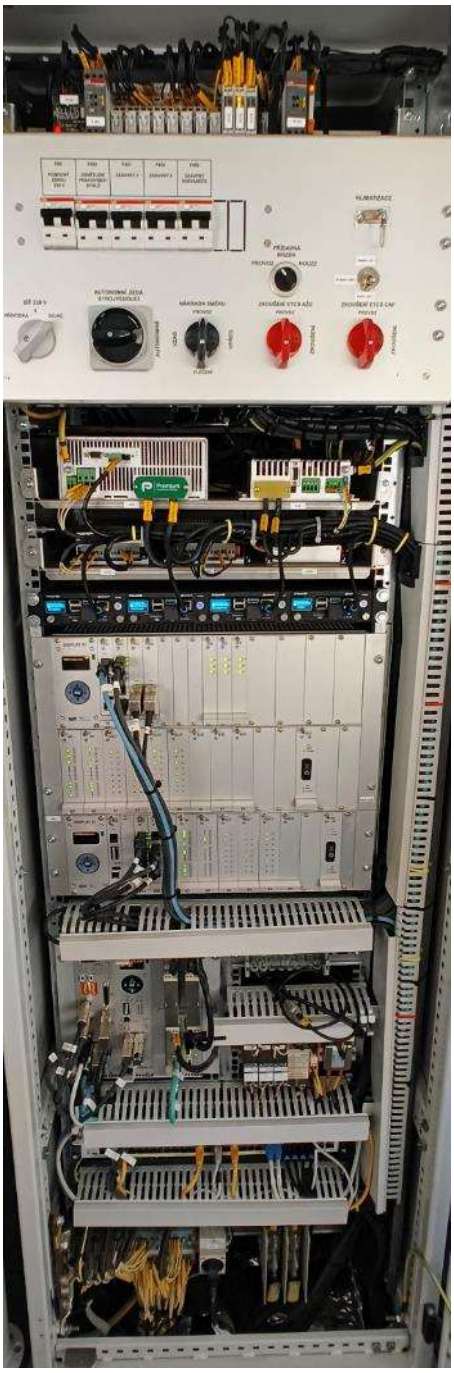

Figure 16: AZD R1 rack containing TCMS, ADM, anti-wheelslip, PAL and REP

ADM

A replacement of driver's hands in terms of driving the train. Based on the data it receives, it leads the train through its journey, automatically and efficiently. Regulates traction and brake of the train to achieve fulfilment of the timetable

(Journey profile) in an energy-efficient way, by taking into account the characteristics of given line description (Segment profile). Built using our 30-year experience with GoA 2 ATO in commercial

Interne

operation, our ADM system is based on our TSI2023-compliant ERTMS/ATO, further extended to support specific requirements of GoA3/4 driverless operation.

Interfaces under test:

- ERTMS Subset 130 and 143 (towards PAL)
- ERTMS Subset 126 and 148 (towards REP) (as C22)

APM

A replacement of driver's brain in terms of task execution, decision making, coordination and monitoring of other systems' function.

Interfaces under test:

- C8 (towards PERception)
- C16 (towards PAL)
- C28 (towards REP)
- C55 (towards TCMS)
- C62 (towards LOC)

PAL

The Protection Adaptation Layer component is in our eyes considered a system, whose main objective is to create a temporary extension of standardized up-to-GoA2 ATO compatible ETCS, to implement new train protection-related functionalities and interfaces to enable development and testing of GoA3 to GoA4 operation. It represents a virtual field for discovery, research, development and testing, to be able to determine all the prerequisites for future GoA3-4-compatible interoperable train protection system. The system is placed between standard ETCS and other key components of an autonomous train. It works both with existing interfaces that are preserved towards GoA2-standardized systems, while at the same time it implements completely new interfaces towards GoA3-4 specific components. In the case of existing interfaces, it can modify existing data as it passes through to or from ETCS, while through new interfaces it acquires and provides key information leading to how to modify the affected existing data to achieve driverless operation.

In simple terms, the system should adapt new automated safety-related functions and behaviours that, in GoA2 or lower grades of automation, would be performed by the driver – e.g. stopping for an obstacle in front of the train. However, the system itself does not make any decisions and is not a direct actor reacting with physical train components such as brakes or traction. It does not have any of its own independent data sources - the system only responds to commands from systems such as APM, and based on these commands adjusts the data flowing towards final actors such as ADM/ATO. According to the previously mentioned example of stopping in front of an obstacle, PAL reacts to information about this obstacle from APM, after which it shortens the speed profile and movement authority of ETCS to the position of this obstacle, while the final actor of this procedure is ADM/ATO, which leads the train to stop in front of this obstacle.

Our PAL implements a specific user interface for better monitoring of its functionality. By drawing important components to the speed/distance graph, like MRSP, EoA, obstacles or TSRs, it is very easy to compare the original MRSP and MA received by ETCS with the one produced by PAL after integrating Obstacles, TSRs and other specifics.

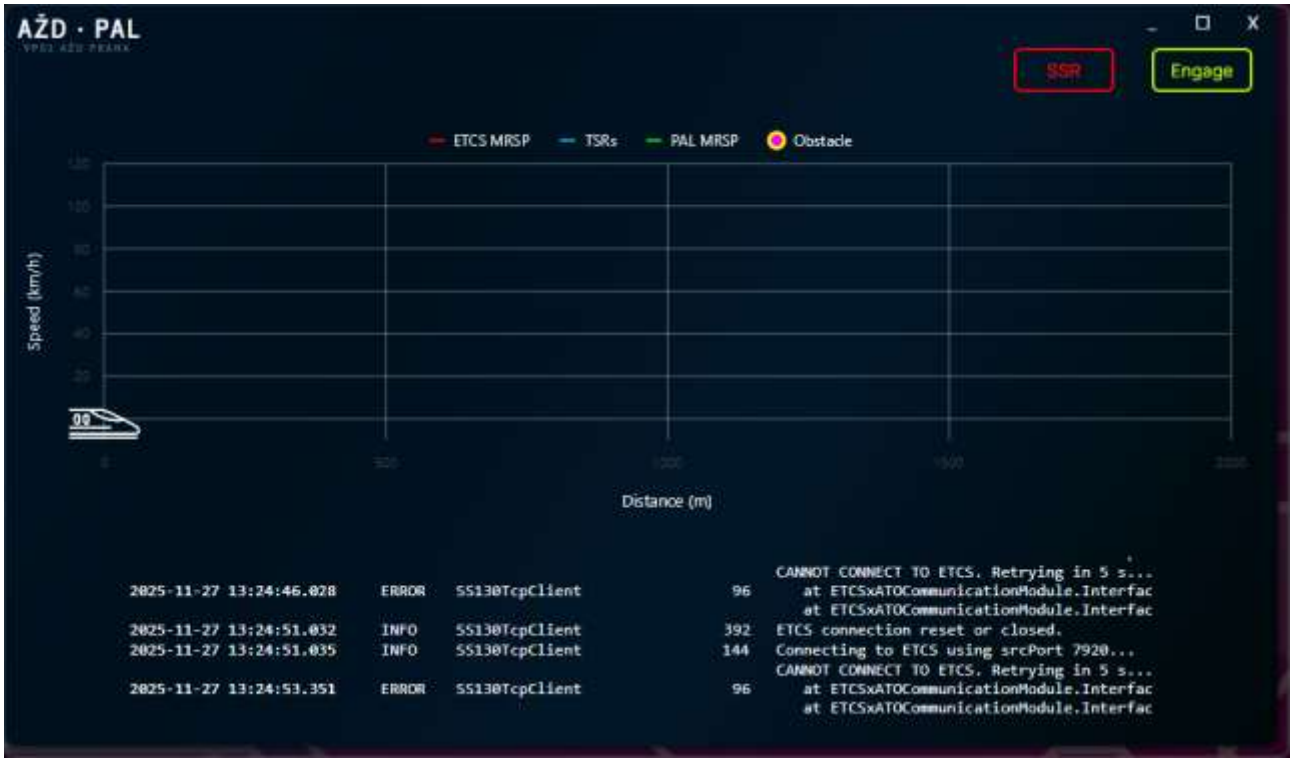

Figure 17: AZD PAL User Interface (unfinished)

Interfaces under test:

ERTMS Subset 130 and 143 (towards ADM)

ERTMS Subset 130 and 143 (towards ETCS-OB)

C5 (towards REPository)

C16 (towards APM)

REPOSITORY

A replacement of driver's memory in terms of data storage. REP also takes care of communication management for other onboard systems, such as the ADM or APM.

Repository process and further provide information exchanged through communication interfaces, stores and provides information from ATO components (on-board and trackside), mediate and monitor mission execution, provide and pre-process the digital map according to the information about the travelled route of the train and its location, provide other additional information from the digital map, and monitor and control the state of itself and the components connected to the repository.

Functions, which are performed by the AŽD repository system:

- Stores the digital map and provides information from the digital map.
    - Creation of micro mapping points based on the received JP.
    - Processing and transmitting track conditions from the digital track map.
    - Processing and transmitting signal locations and descriptions for SCV.

Interne

- Fully functional interception of communication over the SS126 communication interface (mainly includes SP and JP reception and processing).
- Processing MP.
- Autonomous data input to the train.
- Provide information about travelled route to the LOC (micro mapping nodes with segment ID, segment country code and depth in the segment).

Interfaces under test:

- C5 (towards PAL)
- ERTMS Subset 126 and 148 (towards ADM)
- ERTMS Subset 126 and 148 (towards ATO-Trackside over GSM-R connection)
- C28 (towards APM)

PERCEPTION

A replacement of driver's eyes.

The AŽD Perception system consists of three basic parts.

- On-board sensors

Each train head is equipped with a set of sensors that monitor the area in front of the train. The sensors are mounted in special weatherproof pockets. There is a pair of cameras with different focal lengths, two LiDARs with IMU and an infrared camera on each head of train.

- Trackside sensors

An intelligent camera that is installed on the track and is able to inform an approaching train of a possible dangerous situation. This camera is used to monitor the unprotected level crossing.

- Computing unit

A computing machine, which processes all the sensor data, is placed in the switchboard inside the train. This machine is connected to the internal technology network, where communicates with other modules.

Interne

Figure 18: AZD Integration of Perception into the vehicle

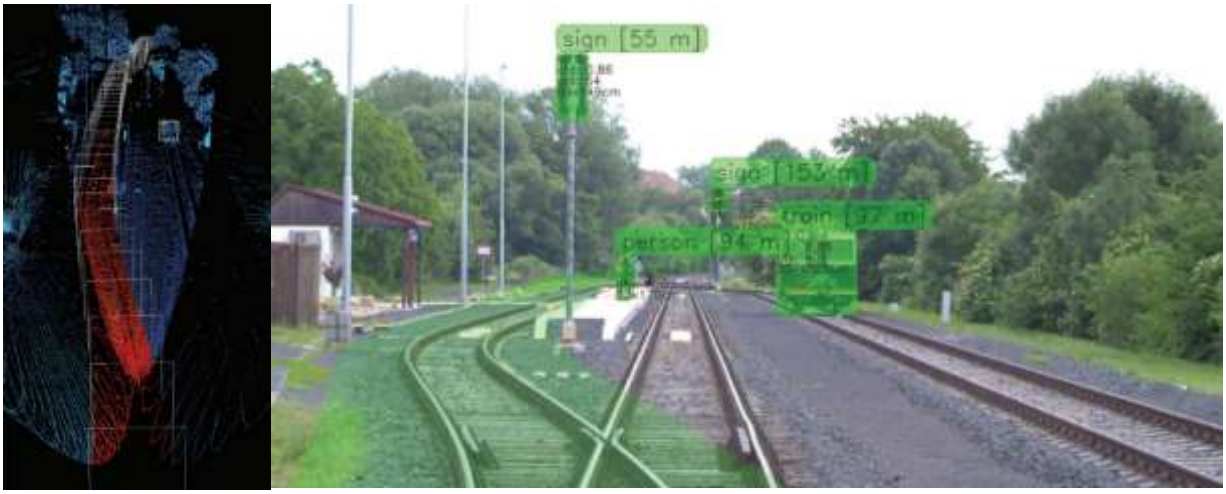

Figure 19: AZD Visualized output of PERception sensoric evaluation of area in front of the train

Interfaces under test:

- C8 (towards APM)
- C62 (towards LOC)

TCMS

The core heart of our testing vehicle. While not being part of GoA3/4 ATO developments, the TCMS is required to perform some additional functions in order to enable testing. While it can not directly allow all autonomous operations to be performed (due to vehicle's physical limitations), some internal simulations have been added (e.g. TCMS internal states) in order to fill the necessary preconditions for setting up the train for GoA3/4 mission.

Interfaces under test:

- ERTMS Subset 139 (towards ADM) – not part of this WP, but necessary
- C55 (towards APM)

2.2.2.2 Interfaces

All of the involved interfaces are physically implemented into all of the systems, while for general testing, all interfaces are provided by the actual systems. No interface simulations have been used (even in laboratory, the simulators perform more functionalities than just emulating the communication interface). Here is the complete overview of all interfaces under test:

1.	**C5**	PAL <-> REP	
2.	**C8**	APM <-> PER	
3.	**C16**	PAL <-> APM	
4.	**C28**	APM <-> REP	
5.	**C55**	APM <-> TCMS	
6.	**C60**	REP -> LOC	
7.	**C62**	LOC -> any system	
8.	**SS126**	ADM <-> REP	+SS148
9.	**SS126**	REP <-> ATO-Trackside	+SS148
10.	**SS130**	ETCS <-> PAL	+SS143
11.	**SS130**	PAL <-> ADM	+SS143

2.2.2.3 Configuration

From the beginning, the configurations were set to emulate the final train installation as much as possible. Therefore, all systems in the laboratory were configured to have exactly the same character and behavior as on the train, including network settings, power, and other characteristics.

Interne

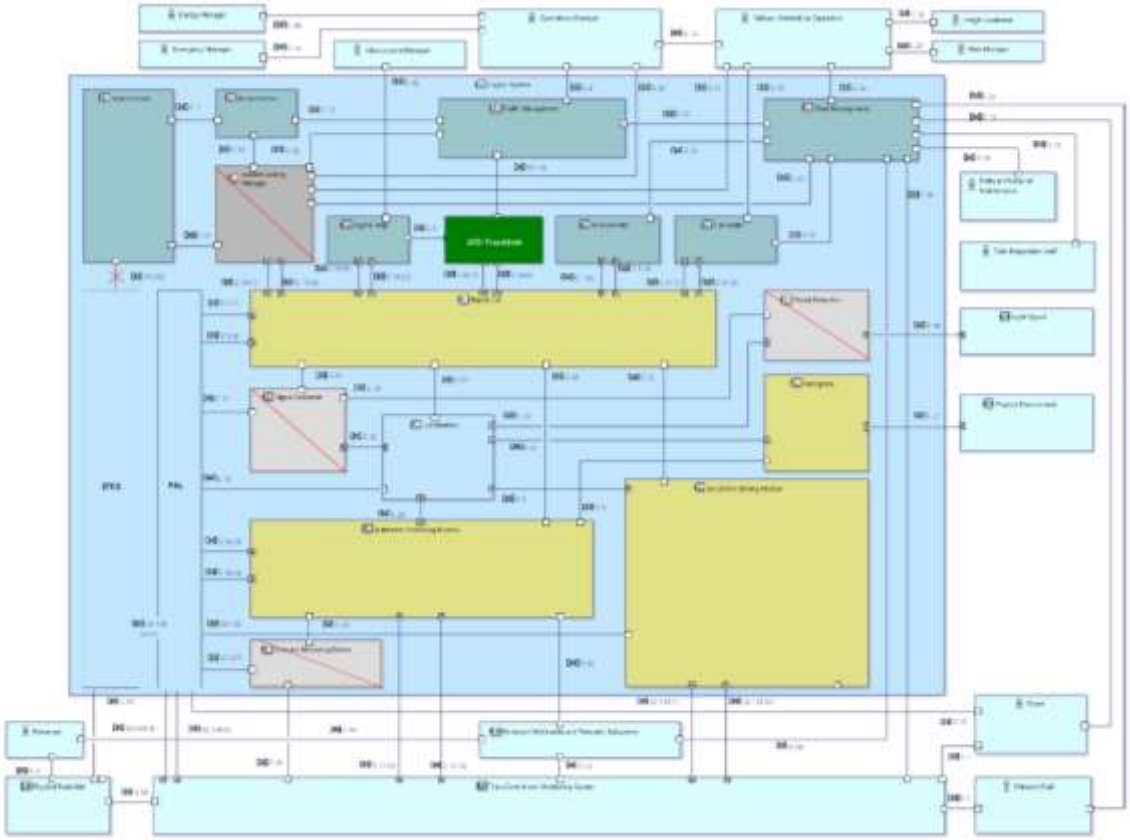

Figure 20: AZD A modified final system architecture used as a reference for testing environment

2.2.2.4 Lab Architecture

The lab architecture was built in order to replicate the architecture of the test train to a maximum possible extent. Only components that needed to be simulated were those, for which it is not possible to bring them to the laboratory – such as the train with TCMS, ETCS-OBU and test environment for PERception testing. As already mentioned, it is not possible to fit the actual train to our laboratory, the harder it is to imagine bringing the whole railway line and its surroundings to it. That means, that major part of PERception testing was done directly on the test train. TCMS has been emulated in a matter to provide exactly the same behavior as the train (even by running the same SW on some modules) – only the movement of the train and the omnipresent physics had to be simulated. The ETCS-OB simulator has been adjusted to replicate ETCS OBU CAF behavior as much as possible, and the ATO-TS emulator has been fed with exactly the same data as the real one used on the Kopidlno line.

2.2.2.5 High Level Test Description

Figure 21: AZD Lab in imagination

Figure 22: AZD Lab in reality

The following WP5 Use-Cases are tested:

- UC 003 – Prepare train for a mission – configure GoA automatically
- UC 005 – Determine and select travelling direction
- UC 006 – Enter ATP data automatically
- UC 010 – Report end of shift / assignment
- UC 012 – Leave train in ready mode
- UC 014 – Perform mission
- UC 019 – Stop at platform for passenger service
- UC 070.2 – Unprotected level crossing
- <No UC> – Obstacle in front of the train detected by PER

The following SRS 1.1.0 functions (the three digits before the function name indicate the corresponding SRs chapter) are tested:

Interne

2.8.1 APM Functions:

- 12.3.1 Determine Running Direction:
  - TESTED: OK
- 12.3.2 Command and supervise horn:
  - TESTED: OK
- 12.3.4 Emulate action from driver:
  - TESTED: OK
- 12.3.8 Define consist leadership:
  - TESTED: OK
- 12.3.9 Determine APM state:
  - TESTED: OK
- 12.3.10 Manage mission execution:
  - TESTED: OK
- 12.3.12 Check departure conditions except signalling:
  - TESTED: OK
- 12.3.13 Define evaluated reaction depending on incident:
  - TESTED: OK
- 12.3.14 Define reflexive reaction depending on incident:
  - TESTED: OK

2.8.2 ADM Functions:

- 12.2.1 Control initial traction effort
  - TESTED AS ERTMS/ATO: OK
- 12.2.2 Determine maximum authorized speed
  - TESTED AS ERTMS/ATO: OK
- 12.2.3 Respect JP Timing Points and Optimize the consumption
  - TESTED AS ERTMS/ATO: OK
- 12.2.4 Stop exactly at the same intended location
  - TESTED AS ERTMS/ATO: OK
- 12.2.5 Compute Operational Speed and Acceleration
  - TESTED AS ERTMS/ATO: OK
- 12.2.6 Acquire train and ADM data
  - Data entry by driver: TESTED AS ERTMS/ATO: OK
  - Data entry by REP: TESTED: OK

Interne

- 12.2.7 Regulate traction and braking
  - TESTED AS ERTMS/ATO: OK
- 12.2.8 Supervise service brake efficiency during operation
  - TESTED AS ERTMS/ATO: OK
- 12.2.9 Request holding brake
  - TESTED AS ERTMS/ATO: OK
- 12.2.11 Check departure conditions
  - TESTED AS ERTMS/ATO: OK
- 12.2.12 Detect that JP has been completed
  - TESTED AS ERTMS/ATO: OK
- 12.2.13 Determine ADM state
  - TESTED AS ERTMS/ATO: OK
- 12.2.14 Manage low adhesion
  - TESTED AS ERTMS/ATO: OK

2.8.3 PER Functions:

- 12.4.1. Determine PER state
  - TESTED: OK
- 12.4.2. Detect animals
  - TESTED: OK
- 12.4.3. Detect railway agents on or along the tracks
  - TESTED: OK
- 12.4.5. Detect humans
  - TESTED: OK
- 12.4.6. Detect infrastructure element
  - TESTED: OK
- 12.4.7. Detect road vehicle
  - TESTED: OK (also people inside car detected)
- 12.4.9. Detect train on adjacent track
  - TESTED: OK (also driver inside cabin of adjacent train detected)
- 12.4.13. Monitor area of interest
  - TESTED: OK
- 12.4.14. Identify obstacle

- TESTED: OK

- 12.4.16. Monitor train front sensors
  - TESTED: OK (loose LAN cable on one LiDAR -> PER [FA] -> stopping the train)
- 12.4.17. Record sensor data
  - TESTED: OK

REP Functions:

- 12.5.1. Acquire JP
  - TESTED: OK
- 12.5.2. Acquire MP
  - TESTED: OK
- 12.5.3. Acquire SP
  - TESTED: OK
- 12.5.4. Acquire Train Parameters
  - TESTED: OK
- 12.5.5. Determine REP state
  - TESTED: OK
- 12.5.6. Manage reporting
  - TESTED: OK
- 12.5.7. Map REP with route
  - Route for complete JP: TESTED: OK
  - Route until End of MA: TESTED: OK

PAL (Train Protection adaptation layer) functions

- 12.6.2. Compute static and dynamic data
  - TESTED: OK
- 12.6.3. Determine ETCS level
  - TESTED: OK
- 12.6.4. Determine ETCS mode
  - TESTED: OK
- 12.6.5. Determine train position and orientation
  - Position is determined by ETCS
- 12.6.6. Handle brake command
  - PAL cannot access the emergency brake nor service brake. The braking is therefore done by ADM – which is respecting the MRSP and MA received from ATP/PAL. On

top of that, PAL implements some additional safety functions. Here is the list of tested functions:

- 12.6.6.A. Shorten MA to position of an obstacle
  - TESTED: OK
- 12.6.6.B. Shorten MA to area of Unprotected LX
  - TESTED: OK
- 12.6.6.C. Shorten MA to start position of TSR if V_TSR = 0 km/h
  - TESTED: OK
- 12.6.6.D. Implement APM TSR to MRSP
  - TESTED: OK
  - Note: Taking into account that PAL always uses the most restrictive speed value for given area, no matter what the source of speed limit (e.g. APM cannot overwrite speed permitted by ETCS).
- 12.6.6.E. Stop the train on APM failure
  - TESTED: OK
  - Example: Failure of some PERception sensor leads to general failure of PER module, which leads to APM failure.
  - Rational: The driverless journey cannot safely continue without PER or APM operational. Therefore, PAL removes MA and sets all permitted speeds to zero, until APM is recovered, which leads to stopping the train.
- 12.6.6.F. Stop the train on connection failure
  - TESTED: OK
  - Rational: PAL cannot safely operate without input information from ETCS (Localization, MA, MRSP) or APM (Obstacles, TSRs). Therefore, it needs to be able to detect irregularities on network connection with ETCS or APM, and remove the MA and permitted speeds, if failure occurs.
- 12.6.6.G. Stop the train on corrupted data input
  - TESTED: OK
  - Rational: If PAL is not able to read data received from ETCS or APM, it needs to stop the train. Since it was not possible to deserialize the input data, the safest approach is to stop immediately. The corrupted data might have contained an obstacle information or MA change.
- 12.6.7. Handle Co-operation shortening of MA
  - This function has been successfully tested in the context of obstacle detection, while not implemented in the context of incident requiring a stop to a rescue point.
- 12.6.8. Manage exchanges with driver
  - Automated Train Data Entry:

Interne

  - TESTED: OK
  - Automated Specific ATO Data Entry:
  - TESTED: OK
  - Engagement of ADM by APM command:
  - TESTED: OK
- 12.6.10. Manage Train Protection parameters
  - TESTED: OK
- 12.6.12. Monitor speed and distance
  - TESTED: OK

2.2.2.6 High Level Description of Test Results

See previous chapter. More detailed information can be added once the testing is complete.

2.2.3 Feedback for Specification and/or Demonstrators

Each point has defined a severity:

- Formal ⓘ (lowest, mostly related to errors or inaccuracies in specification)
- Serious ⚠ (severe – has negative impact on functionality of autonomous GoA3-4 operation)
- Danger! ☠ (most serious – may lead to dangerous situation and requires immediate solving)

2.2.3.1 General Specification Feedback

Object positioning within line - distance from Balises vs. Odometry

Severity: DANGER! ☠

Problem: The specification presumes multiple object positions to be defined as a distance from last read balise group in decimeters.

Rational: Such a definition is highly unusual within established ETCS and ATO practices. In the ETCS/ATO domain, all spaces and objects are defined using an odometric position within an odometric reference space. This means that the standard reference for the train position—and for all elements located behind or ahead of the train—is an odometer, or more precisely an odometric value, typically expressed with a resolution of one centimetre (cm).

There appears to be no rational justification for adopting different practices for the GoA3/4 interfaces, where the positions of elements, including the Movement Authority (C5), Obstacles (C16), and others, are transmitted as distances from a balise in decimetres. This approach imposes additional computational steps on safety-critical components such as the ATP. In order to compute the odometric position of an element, the ATP must first locate the relevant balise and determine its position, and only then compute the resulting odometric value to incorporate it into the MRSP, MA, and other related fields.

Furthermore, this methodology requires assuming that the ATP maintains knowledge of multiple previously read balises—potentially more than two—in scenarios where the received information

does not refer to the last read balise (or balise group) but instead to one encountered earlier in the past.

A certain level of risk arises when a balise group is passed only a few milliseconds after an on-board obstacle detection. In such a situation, the APM would initially assign the detected obstacle to the previously read balise group. When the balise group changes, the APM should, in principle, recalculate the obstacle's position so that it is referenced to the newly read balise group.

However, several questions must be considered:

• Is it guaranteed that the APM will always perform this recalculation correctly?

• Is it certain that the ATP will still retain information about balise groups read in the past?

• After the recalculation, can we be sure that the detected obstacle will still be interpreted as the same obstacle?

• What happens if multiple obstacles are detected at the same position, each detected just after a different balise was last read?

A whole new problem is raised when there is a Trackside obstacle detector and APM is informed about such obstacle, which may be several kilometres in front of the train. APM has no knowledge about how far this is from the train's last read balise group. APM cannot wait, until the LRBG is the latest one in front of the obstacle – the obstacle may be only a few meters past such balise group. APM has no knowledge about balises whatsoever, except the one that has been last read. Here, an APM making comparison of information received from the Trackside OD and the Digital Map to recalculate the obstacle's position as an odometric value would be much more efficient.

Conclusion: If all components are aware about the train's odometer, the situation would be much easier and safer for every system involved in the railway operation.

Proposal: All positions within 1D railway linespace description should follow the ETCS standards and be computed and provided as a resulting odometric position with the possible resolution of centimeters.

2.2.3.2 ADM-Related Feedback

ADM definition

Problem: ADM is defined as a state machine in the documentation. ADM/ATO is a system too complex to be a simple state machine – but obviously, it may act like one for the surrounding systems.

Proposal: ADM should not be referred to as a "state machine".

ADM/ATO behaviour after stopping the train outside stopping points

Severity: DANGER! ☠

The following feedback is applicable for ADM, but to some extent also for GoA2 ERTMS/ATO:

- Context 1: On stopping points, ADM/ATO, when being in SE/AV state, checks for various conditions to be able to enter Ready (RE) state, such as:
    - Doors are closed
    - The Time of Departure has passed

- Train Hold is not active
- Movement Authority is granted (for reasonable length of journey ahead)

- Context 2: When ADM/ATO goes into REady state, it signals that it has all the conditions fulfilled for start moving the train. In simple terms, when ADM/ATO is REady, it should positively react to engagement command (given either by Driver or APM).
- Context 3: When ADM/ATO stops the train outside Stopping Point / Platform (for whatever reason – mostly for stopping before End of Movement Authority (EoA), it should stay Engaged (in EG state). After the MA is extended, it is expected to automatically start moving the train again, without any interaction with the Driver (GoA2) or APM (GoA4 through ATP).
- Problems:
  - For both GoA2 and GoA4, There is absolutely no confirmation, that it is safe to start moving the train again.
  - For GoA2, the train starts to move, but driver may not be vigilant at the moment. An obstacle very close to front of the train may be hit (since no PER is used for GoA2).
  - For GoA4, systems interacting with the ADM/ATO can not recognise from ADM/ATO state, whether the train is moving and not. Also, there is absolutely no feedback from ADM/ATO, what is its momentary reason for the train being stopped. Systems like APM may find this information useful.
  - For GoA4, In stopping points, there is a specific sequence performed before the train can start moving. APM waits for ADM to be REady, and then, it checks ist own conditions, whether it is safe to engage the train (e.g. no obstacle is in front of the train). However, such sequence is completely missing, after the train has stopped outside stopping point. Why?
- Proposal:
  - An Engagement sequence should be performed after EACH STOP of the train, no matter the reason. Both ADM/ATO and APM should always check conditions, that it is safe to proceed with the journey. Therefore, ADM/ATO should loose Engagement after stopping, again check conditions, and only after conditions are fulfilled (e.g. MA Extended), it should fo into Ready again. After that, it waits for Engagement command from Driver or APM.

Naming

Severity: FORMAL ⓘ

Problem: The documentation is filled with typos or accidental substitution of “ADM” and “APM”

Rational: The rationale for renaming ATO-OB to ADM for GoA3-4 is unclear. This change introduces a significant risk of misspelling, misunderstanding, or mistyping, and it also overrides the long-established terminology associated with Automatic Train Operation. Since ADM is fundamentally built upon the ATO framework, there appears to be no justification for such a renaming. Maintaining the ATO-OB designation for GoA4 would be far more appropriate and would preserve consistency.

It would, for example, be preferable to state that “the ATO system, from a certain baseline onward, also supports autonomous operation in GoA3 and GoA4 modes.” Alternatively, the system could simply be referred to as “GoA2 ERTMS/ATO” and “GoA4 ERTMS/ATO.”

While it is understood that additional components—such as REP and ADM—are expected to become part of the broader “ATO” concept in the future, this evolution risks introducing confusion and blurring the distinction between ATO systems compatible only with GoA2 and those capable of supporting GoA4, even when currently operating in GoA2 mode.

Proposal: Keep ADM referred to as “ATO-OB”.

ADM Behavior for stopping to EoA

Severity: SERIOUS ⚠

See chapter 2.2.3.3 - MA shortening due to obstacles

2.2.3.3 ATP-Related Feedback

MA shortening due to obstacles

Severity: SERIOUS ⚠

Problem: ATP should use the same MA for both signals and obstacle EoAs.

Objection: It would be strongly recommended to introduce a clear distinction between the two types of Movement Authority: the Signal MA and the Obstacle MA.

Rational: The ADM may implement slightly different behaviour for braking to a signal EoA and obstacle EoA. ADM should be aware, whether it is leading the train to a stop for a red signal or to stop for an obstacle. The braking mechanism for Obstacle should be slightly different (e.g. more steep braking curves calculated). Also, for GoA4 development and testing purposes, but also for GoA4 ADM running in GoA2 with obstacle detection at the moment, ADM should have the option to inform the Driver (or a researcher) that it is braking to an Obstacle, not just any EoA. Some ADM/ATO manufacturers or some vehicles have a specific ATO DMI screen to inform the driver, what the system is doing (why is the train currently braking? With ETCS DMI only, the driver has no direct clue).

Proposal: Introduce a specific EoA for Obstacles to the ATP or, add the option to inform ADM whether the EoA is related to a signal or an obstacle.

Additional reasons for stopping the train

Severity: SERIOUS ⚠

For additional reference, see also chapter 2.8.5, function 12.6.6.

Problem: ATP lacks the options to stop the train emergencily for a reasons different than an obstacle

Rational: APM and ATP should have the option to request/perform stop of the train immediately for emergency reasons (different than an obstacle), such as:

- Remote controller’s request
- Any onboard systems’ failure

- Malformed data detection
- Key system disconnection / unresponsiveness
- Etc.

Proposal: Introduce another packet and specification for enabling the APM and ATP to stop the train immediately for any reasons other than an obstacle. Or, add a specific type of TSR to be provided by APM to ATP, with a zero speed meaning the need of immediate stopping. Also, it should be distincted, whether ATP should use full service brake or emergency brake to stop the train.

LRBG definition

Severity: FORMAL ⓘ

Problem: NID_REFBALISE is intentionally discarding N_PIG variable

Rational: According to SRS1.1.0 Appendix A, on C16 interface, the definition of Last Read Balise Group (LRBG) was unified with the one according to SS130's "NID_REFBALISE". That itself is a very pleasant advancement in comparison to SRS 1.0.0. On the other hand, it is purposedly ignoring the N_PIG variable, which indicates which balise of the balise group should be used as a reference. That may bring some unexpected behaviour to the system, for cases, when only part of a Balise Group has been read. From our point of view, there is no strict need for discarding the N_PIG variable, since ATP is aware about last two balises read.

Proposal: Include the N_PIG variable back to NID_REFBALISE, or use Runover information instead.

ATP specification

Severity: FORMAL ⓘ

Problem: Missing Projects / WorkPackages / WorkGroups to specify ETCS support for GoA3-4 autonomous operations.

Rational: There is currently no workgroup that would specialize on developing the ETCS specification supporting autonomous train operations. Multiple topics need to be solved:

- Autonomous operations with ETCS in other modes than FS / AD
- Autonomous shunting done by ADM under ETCS supervision
- Etc.

2.2.3.4 APM-Related Feedback

Missing option to control whistle or more horns

Severity: SERIOUS ⚠

Problem: APM currently provides an option to control one horn only. It lacks the option to control more horns and a train whistle.

Rational: Some trains have more horns (more tones) and/or a whistle. APM has no option to distinguish between whistle and horn. For obstacles, it would be more appropriate to use horn. But, a whistle may be often used for less-emergency situations, e.g. start of shunting movement, start of (any) movement on the train, preliminary warning for humans close to the tracks etc.

Proposal:

- Add the option to C55 for APM to control whistle
- Add to C55 the option to use iterations for n-number of horns of given train. TCMS should inform APM of how many horns is available.

Missing option to control train's light signalling

Severity: SERIOUS ⚠

Problem: APM can not specifically control the train's external lighting.

Rational: At various countries, a train's lighting (combination of positions of lights in different colors) have a defined meaning. For example, in Czechia, different light combination is used for running train's start (three white lights) and end (two red lights), shunting (two white lights on both sides), stop by any means necessary (left red, upper white, right white) etc. etc. APM has no means of controlling train's external lighting.

Proposal: Add the option to C55 for APM to control every train's light lamp and its colour separately.

Missing option to control high beams / reflectors

Severity: SERIOUS ⚠

Problem: APM lacks the option to control (switch on / off or dim) high beams

Rational: An option and a methodology for controlling low beams / high beams on train external lighting needs to be solved. The use of high beams is necessary at night – while it may also increase the efficiency of PERception module at night sceneries with object classification. Methodology is necessary, because the use of high beams is limited by legislation. For example, the high beams need to be turned off for passing oncoming trains, or even on level crossing in order not to dazzle the drivers. A cooperation with PERception module is necessary.

Proposal: Add the necessary fields to related interfaces and create a methodology for controlling high beams autonomously.

Mission profile management and other issues

Severity: FORMAL with utmost importance ⓘ

Problem: It is not clear what the MP management should be in several cases:

- Can a MP or a specific Task be updated / rewritten?
- What to do with Unavailable or Invalid Tasks? Can a task be invalid or unavailable?
- SRS interface description sometimes refers to a following condition:
  - If (M_TASK_TYPE == TRJ). Such Task type does not exist.
- What if other Task needs to be performed during an active TJP task? e.g.:
  - Turn on/off compartment lights before/after entering/leaving a tunnel

Proposal: To solve this problem.

2.2.3.5 REP-Related Feedback

Micromapping extension

Severity: FORMAL ⓘ

Problem: Currently, after introducing a C5 interface and its ETCS_REP_MOVAUTHORITY packet, it is presumed, that REP would generate micromapping for REP and PER starting from the LRBG and ending at the EoA position.

Rational: Although this does not create an issue, they question the need to limit the Micromapping to such an extent. They consider that the option to provide Micromapping for the entire length of the received Journey Profile or Segment Profile should be included.

Proposal: Add the simpler option to provide Micromapping for the whole available JP/SP.

2.2.3.6 TCMS-Related Feedback

TCMS Cabin Status

Severity: SERIOUS ⚠

Problem: A CAB status differentiating between “CAB closed, GoA4 inhibited” and “CAB closed, GoA4 granted” is missing.

Rational: In SRS 1.1.0. Appendix A, C55 description, a whole new description is introduced for M_CAB_STATUS variable, which, for a state where all cabins of the train are closed, does not differentiate whether it is possible for the CAB to be opened by APM or not is missing (compared to SRS 1.0.0.). That creates a huge problem for other system’s knowledge of whether the train is ready for GoA4 operation or not. It is a rational approach for transition between GoA<=2 to GoA4, for the cabin to be firstly closed by driver, then for GoA4 to be enabled, and only then to be available for APM to open the cabin. The reason is very simple: The APM cannot overrule a train currently controlled by the driver. It needs to wait for a state, where taking control of the cabin is enabled.

Proposal: Put two states for CAB closed: “GoA4 inhibited” and “GoA4 granted/available”.

Interne

2.3 CAF

2.3.1 Methodology

Tests have been defined and designed based on the functions and requirements described within the scope of R2DATO WP6 task 6.5 deliverable document, with additional reference from task 6.1 deliverable document use cases. The validation of the prototypes is made through tests presented in the document that have been defined to cover implemented functions and requirements for Automatic Processing Module (APM) and Automatic Driving Module (ADM). To ensure the objectivity of the process, the test definition and prototype implementation has been done by separate groups.

After the test implementation phase, these tests are executed on a test environment to validate the prototypes. The validation process has been structured into three separated stages:

1. **PC environment:** SW Validation in a simulated environment with unit under test running on PC (TRL 4). Interfaces are fed through simulated physical layer and HW is emulated. PC environment has two stages: one subsystem level test where unit under test is a single subsystem and other subsystems are simulated and integration test stage where SW for subsystems are deployed together by enlarging system boundaries.
2. **Target Environment:** SW/HW Validation where system boundaries are simulated and unit under test is running on target HW in a HIL setup (TRL 5). Interfaces are fed through real physical layer.
3. **Demonstrator Environment:** Demonstrator Validation where system boundaries are real (TRL 6) including all interfaces. This environment allows to demonstrate the integration among components. Information about this validation stage is inside WP38 scope. Demonstrator was carried out in collaboration with NS (Nederlandse Spoorwegen) In a CAF Civity train from SNG fleet in a 10km track in the Netherlands containing stations and shunting yards.

The test results have been used to verify the correct operation of the prototypes, enabling iterative implementation whenever issues or anomalies were detected.

APM and ADM are planned to be validated within the three phases. Phase 3 includes modified versions for other required subsystems: TCMS, ETCS (Adapter), LZ. based on previous proprietary baseline, as boundary systems in demonstrator phase are not fully GoA3/4_1.1.1 baseline compliant due to retrofit constraints, third phase of validation is out of scope of this document as it is not focused on current GoA3/4 baseline validation. Results for demonstrator are briefly described in the document and furtherly described in WP38's D38.4.

The objective of this document is not validating the integration of the prototypes, as this activity has been already carried out within the framework of WP38 Task 38.4 Demonstrator. The focus is placed on the functional validation of the individual components through specific test scenarios.

The test execution generates some output files containing both the expected and actual results, as well as the data packets that have been exchanged between interfaces. These outputs are used for checking the correct behavior of the prototype and for performing a detailed analysis of the data in case of being necessary. The output is used also for generating the test result classified as validated, failed or justified validation (fulfilment justified) with reasoning for compliance.

Interne

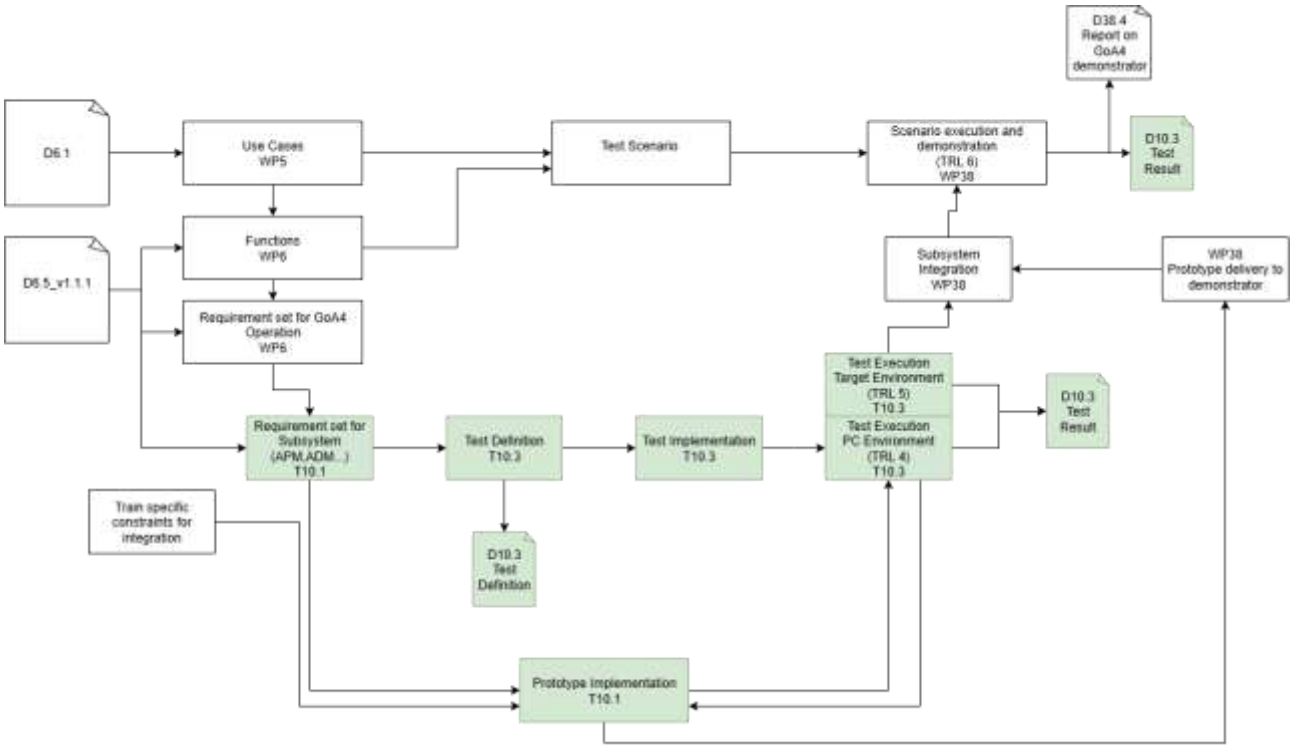

Figure 23: CAF methodology diagram

Diagram shown in Figure 23 describes the methodology used for implementation and further validation in addition to input and output documents inside R2DATO structure.

2.3.2 Test Activities Description

All the high level descriptions and test reports included in this section are summaries extracted from several internal CAF documents. Specifically: ADM_alcance_test_plan_y_report_v_1.1.1.xlsx [2], which contains the complete test definitions, scope, and test report for the ADM subsystem; APM_alcance_test_plan_y_report_v_1.1.1.xlsx [3] for the APM subsystem; and Alcance_y_descripción_general_testing_v_1.1.1.docx [4] for the overall testing scope and general definitions.

2.3.2.1 Equipment Under Test

The equipment used during the testing phase includes prototype HW units for validation, simulators for generating customized input signals, Hardware in a Hardware in the Loop setup (TRL 5), proprietary testing software and dedicated testing PCs.

The testing environment is designed to support unitary prototype validation or integration between prototypes depending on applied configuration. As the main objective is the validation of the prototypes, this equipment and configuration are going to be the focus.

On target testing environment, the prototypes run on dedicated target hardware on laboratory setups prepared to replicate onboard conditions. These environments utilize hardware components used in the train.

Interne

Figure 24 illustrates a TRL5 level testing environment. Sections 1 and 2 (highlighted in green) represent onboard equipment installed on the train, where the prototypes and adapters are executed. Section 3 (highlighted in red) corresponds to the PC hosting the test engine, which manages the execution of test scenarios, needed simulators and collects diagnostic data.

Figure 24: Target environment HW setup

The test execution is managed by proprietary testing software that manages the interface simulators, testing engine and validation software, all of them hosted in a PC environment. This architecture and process is explained in chapter 2.3.2.4 Lab architecture.

The prototypes that have been developed within the scope of WP10 task 10.1 are the following:

ADM: The Automatic Driving Module GoA3/4 v1.1.1 Delta has been developed based on 2023 TSI for ATO GoA2 version.

Interne

APM: The Automatic Processing Module, based on the D6.5 GoA34_R2D_1.1.1 specification document.

Adapter: It addresses the limitations of our TP system while also enabling localization and signal conversion functionalities. The interfaces with TCMS, REP and TP are internal, while with APM and ADM are the ones defined in the STS: C29, C16 and SS130.

2.3.2.2 Interfaces

Testing environments for APM and ADM subsystems are based on simulators for all the relevant input/output interfaces that are fed from a testing engine which reads a Scenario File. Testing engine reads the outputs of relevant interfaces to verify the expected behavior. Simulated interfaces are:

1. **C55 APM-TCMS**: Simulate TCMS inputs with external simulator for linear dynamics affecting positioning or speed.
2. **C8 APM-PER**: Simulate PER detections for obstacle/track conditions and move their position based on external dynamics simulator.
3. **C16 APM-ETCS**: Simulate ETCS behavior in a simple way to generate inputs for APM, dynamics are linked to external simulator.
4. **C28 APM-REP**: Simulate REP behavior with data stimulus from testbench file.
5. **C29 LZ-APM**: Simulate Global positioning based on scenario definition and external dynamics simulator.
6. **C65 APM-ORD**: Check outputs from APM for juridical recording. (Out of scope for implementation)
7. **C63 APM-OMTS:** Check interaction between APM and On Board Multimedia and telematic subsystem. (Out of scope for implementation)
8. **C22 ADM-REP**: Simulates REP data input/output from testbench file.
9. **SS130 ADM-TP**: Simulates TP behavior with external positioning simulator.
10. **SS139 ADM-TCMS**: Simulates TCMS inputs with external simulator for linear dynamics affecting positioning or speed.

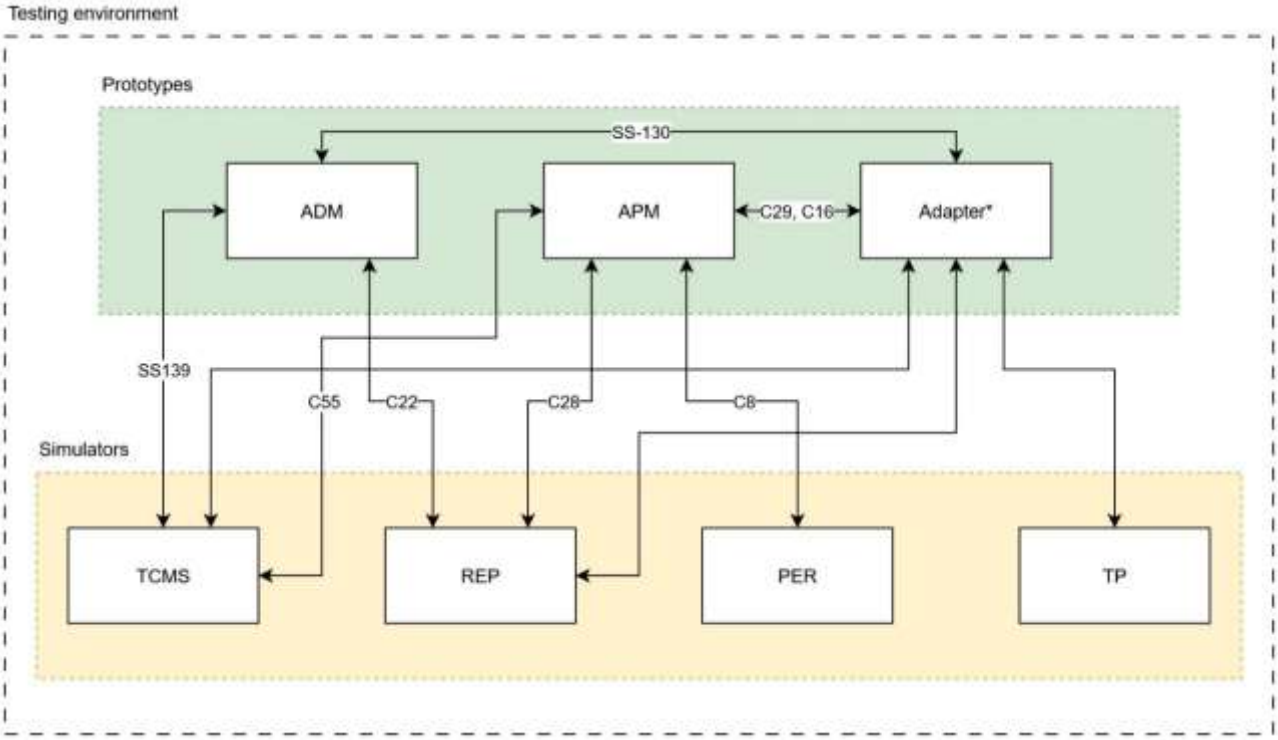

Figure 25: Developed prototypes and interfaces

Figure 25 Shows a general overview of the prototypes that have been developed and the simulated interfaces.

2.3.2.3 Configuration

Three distinct configurations have been employed during the validation process:

Local configuration: In this setup, both prototypes and simulators are executed locally in a PC environment, without the involvement of any external hardware. This configuration is primarily used for early-staged functional validation and debugging.

Target configuration: Prototypes are deployed on actual onboard hardware, configured to run with specific train IP addresses. While some interfaces are simulated, this setup allows realistic operational context.

Train configuration: Same configuration as TRL5 environment adding real boundary systems.

It is important to also note that each testing environment can be configured to accept more than one prototype to test the integration.

2.3.2.4 Lab architecture

The logical architecture for laboratory validation is identical on PC environment (TRL 4) and Target Environment (TRL 5).

The architecture illustrated in the figures below (Figure 26, Figure 27) represents the validation setup for APM and ADM, referred to as unit under test. These setups are designed to simulate operational conditions and verify the correct behavior under controlled stimuli monitoring the responses.

The test engine orchestrates the validation process by executing QML File Scenarios, which defines a sequence of inputs and expected output steps.

Each simulator supports bidirectional communication, allowing both input packets to be sent to the Unit Under Test and output data to be received for analysis.

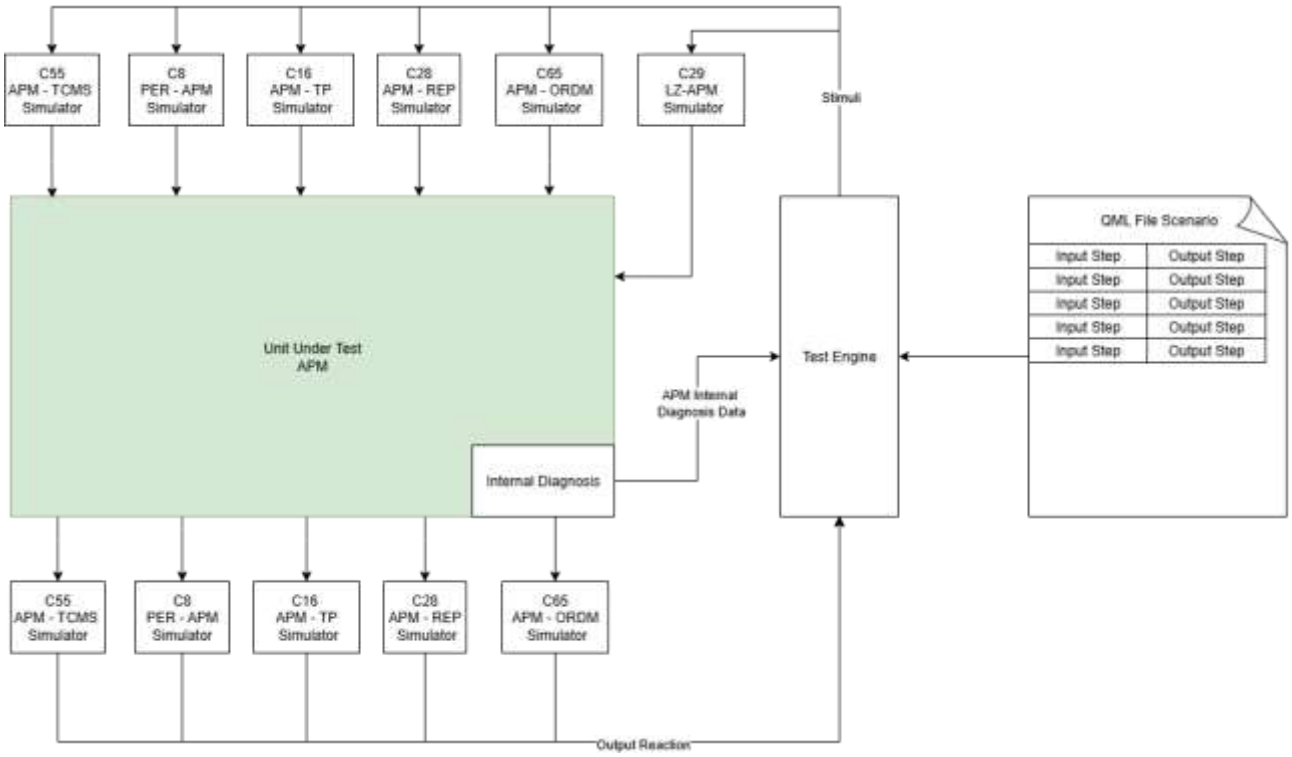

Figure 26: APM validation architecture

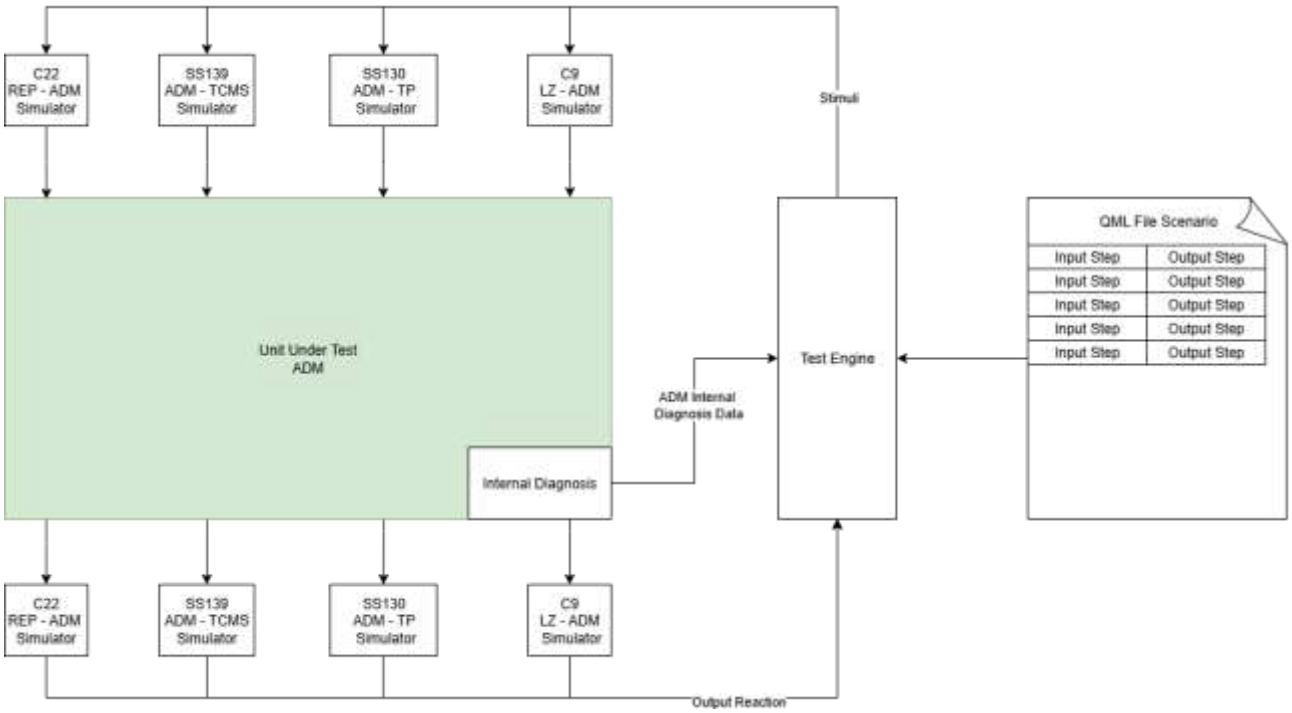

Figure 27: ADM validation architecture

The test engine is also capable of simulating position and speed with an external simulator for linear dynamics calculating it depending on the outputs received by the unit under test.

2.3.2.5 High level test description

After analysing WP6 D6.5 version 1.1.1 and reviewing all the functions developed in accordance with the deliverable. The following tests have been implemented to validate the developed functionalities and requirements:

2.3.2.5.1 APM Functions:

- **12.3.1 Determine Running Direction:**
  - Test TC-CAF-APM-001_1: Activation of cabin when Unit is leading.
  - Test TC-CAF-APM-001_2: Non-Activation of cabin when Unit is non leading.
  - Test TC-CAF-APM-002_1: Next journey direction is the same as last one. Test correct cabin activation.
  - Test TC-CAF-APM-002_2: Next journey direction is the opposite to last one. Test correct cabin activation.
- **12.3.2 Command and supervise horn:**
  - Test TC-CAF-APM-003_1: Horn is commanded by APM in predefined location.
  - Test TC-CAF-APM-003_2: Horn is commanded in presence of applicable incident.
  - Test TC-CAF-APM-004: Horn defect detected by TCMS is reported.

Interne

- **12.3.3 Deactivate driver activity control:**
  - Test TC-CAF-APM-005: Test APM deactivates vigilance system when APM state is GoA3 or GoA4.
- **12.3.4 Emulate action from driver:**
  - Test TC-CAF-APM-001_1: Activation of cabin when Unit is leading.
  - Test TC-CAF-APM-006: Direction controller remains neutral until departure is allowed.
  - Test TC-CAF-APM-007: Direction controller is neutral when APM state is AV.
  - Test TC-CAF-APM-008: APM emulates EG button if all conditions are fulfilled.
- **12.3.5 Manage supervision orders:**
  - Test TC-CAF-APM-006: APM inhibits train departure until it is authorized.
  - Test TC-CAF-APM-009: ES request is forwarded to TP by APM when requested from trackside.
- **12.3.6 Start coupling:**
  - Test TC-CAF-APM-010: Test that passive coupling starts when requested by the mission execution and holding brake is applied.
  - Test TC-CAF-APM-011: Active coupling starts when it is authorized and requested by mission execution.
  - Test TC-CAF-APM-012: Not request coupling mode to TCMS if the speed is not equal or higher to configurable speed.
  - Test TC-CAF-APM-013: APM reports coupling status to REP.
- **12.3.7 Start splitting:**
  - Test TC-CAF-APM-014: Train is standstill when splitting.
  - Test TC-CAF-APM-015: Splitting status is received by leading TCMS.
- **12.3.8 Define consist leadership:**
  - Test TC-CAF-APM-016: Tests that when the consist is not leading, system states are determined, mission execution is managed, and MP is acquired.
  - Test TC-CAF-APM-017: APM closes cabin before changing consist leading status.
- **12.3.9 Determine APM state:**
  - Test TC-CAF-APM-018: Cabin is activated by driver in GoA1/GoA2 and by APM in GoA3/GoA4.
- **12.3.10 Manage mission execution:**
  - Test TC-CAF-APM-019_1: Train is at standstill when updating the MP.
  - Test TC-CAF-APM-019_2: Train is moving when updating the MP, but does not affect current task.

- Test TC-CAF-APM-016: Check that all systems are executing when the consist is not leading. But commands are not executed apart from splitting when TUP.
- Test TC-CAF-APM-020: APM reports mission and task status.
- Test TC-CAF-APM-021: Checks that SoM is triggered when TCG conditions are fulfilled and GoA3/4 is requested by RU.
- Test TC-CAF-APM-022: TCL is tarted when there are no passengers in the train.
- Test TC-CAF-APM-023: APM monitors tasks when state is GoA1/GoA2.
- Test TC-CAF-APM-024: Test that APM waits for cabin activation when TCG task changes APM state from GoA3/4 to GoA1/2.
- Test TC-CAF-APM-025: TCG and TCR tasks can be executed in parallel when TJP is ongoing.
- Test TC-CAF-APM-026_1: APM terminates mission when the last task is TST type shutting down and TCMS is in shutdown mode.
- Test TC-CAF-APM-026_2: APM terminates mission when it is operating in GoA1/GoA2 and the last MP task is completed.

- **12.3.12 Check departure conditions except signalling:**
  - Test TC-CAF-APM-006: APM inhibits train departure until it is authorized.
- **12.3.13 Define evaluated reaction depending on incident:**
  - Test TC-CAF-APM-027: Report to REP if an object of type (CLOO, HUMN, ANIM) is detected in the rail system environment.
  - Test TC-CAF-APM-028: Direction controller is set to neutral and requests RAP.
  - Test TC-CAF-APM-029: APM reports to trackside if GoA3/GoA4 is inhibited while being in GoA3/GoA4.
- **12.3.14 Define reflexive reaction depending on incident:**
  - Test TC-CAF-APM-030_1: Obstacle EOA is sent to ETCS when an obstacle is detected within the train gauge.
  - Test TC-CAF-APM-030_2: Obstacle EOA sent to ETCS clears when the obstacle disappears.
  - Test TC-CAF-APM-031: Direction controller is set to neutral after a reflexive reaction.
  - Test TC-CAF-APM-032: APM reports coupling information during coupling operation.

2.3.2.5.2 ADM Functions:

- **12.2.1 Control initial traction effort**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.2 Determine maximum authorized speed**
  - SS125 Requirements, already validated in GoA2 2023 TSI.

Interne

- **12.2.3 Respect JP Timing Points and Optimize the consumption**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-001: When a JP is received, if the train is stopped within the range of a TP it marks as reached, otherwise selects the next TP.
- **12.2.4 Stop exactly at the same intended location**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.5 Compute Operational Speed and Acceleration**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.6 Acquire train and ADM data**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-002: Checks that the train data, this is, TCMS and REP data is received by ADM.
- **12.2.7 Regulate traction and braking**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-003: ATO-OB does not move the train when virtual cab is activated until departure conditions are fulfilled.
- **12.2.8 Supervise service brake efficiency during operation**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.9 Request holding brake**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.11 Check departure conditions**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.12 Detect that JP has been completed**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.13 Determine ADM state**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-005: Tests the JP applicable conditions for ATO Operational.
- **12.2.14 Manage low adhesion**
  - SS125 Requirements, already validated in GoA2 2023 TSI.

A more detailed specification of the tests is included in 2.3.2.6 High level description of test results: further detailed elaboration along with a traceability matrix for both APM and ADM.

Interne

2.3.2.6 High level description of test results

A high-level description of TRL4 and TRL5 test results along with their corresponding functions is presented below, while the results of TRL6 tests are provided below in the Table 1 and Table 2 with a more detailed analysis.

2.3.2.6.1 APM Functions:

- **12.3.1 Determine Running Direction:**
  - Test TC-CAF-APM-001_1: Test passed. Validated.
  - Test TC-CAF-APM-001_2: Test passed. Validated.
  - Test TC-CAF-APM-002_1: Test passed. Validated.
  - Test TC-CAF-APM-002_2: Test passed. Validated.
- **12.3.2 Command and supervise horn:**
  - Test TC-CAF-APM-003_1: Test passed. Validated.
  - Test TC-CAF-APM-003_2: Test passed. Validated.
  - Test TC-CAF-APM-004: Test passed. Validated.
- **12.3.3 Deactivate driver activity control:**
  - Test TC-CAF-APM-005: Test passed. Validated.
- **12.3.4 Emulate action from driver:**
  - Test TC-CAF-APM-001_1: Test passed. Validated.
  - Test TC-CAF-APM-006: Test passed. Validated.
  - Test TC-CAF-APM-007: Test passed. Validated.
  - Test TC-CAF-APM-008: Test passed. Validated.
- **12.3.5 Manage supervision orders:**
  - Test TC-CAF-APM-006: Test passed. Validated.
  - Test TC-CAF-APM-009: Test passed. Validated.
- **12.3.6 Start coupling:**
  - Test TC-CAF-APM-010: Test passed. Validated.
  - Test TC-CAF-APM-011: Test passed. Validated.
  - Test TC-CAF-APM-012: Test passed. Validated.
  - Test TC-CAF-APM-013: Test passed. Validated.
- **12.3.7 Start splitting:**
  - Test TC-CAF-APM-014: Test passed. Validated.
  - Test TC-CAF-APM-015: Test passed. Validated.
- **12.3.8 Define consist leadership:**
  - Test TC-CAF-APM-016: Test passed. Validated.

Interne

- Test TC-CAF-APM-017: Test passed. Validated.

- **12.3.9 Determine APM state:**
  - Test TC-CAF-APM-018: Test passed. Validated.
- **12.3.10 Manage mission execution:**
  - Test TC-CAF-APM-019_1: Test passed. Validated.
  - Test TC-CAF-APM-019_2: Test passed. Validated.
  - Test TC-CAF-APM-016: Test passed. Validated.
  - Test TC-CAF-APM-020: Test passed. Validated.
  - Test TC-CAF-APM-021: Test passed. Validated.
  - Test TC-CAF-APM-022: Test passed. Validated.
  - Test TC-CAF-APM-023: Test passed. Validated.
  - Test TC-CAF-APM-024: Test passed. Validated.
  - Test TC-CAF-APM-025: Test passed. Validated.
  - Test TC-CAF-APM-026_1: Test passed. Validated.
  - Test TC-CAF-APM-026_2: Test passed. Validated.
- **12.3.12 Check departure conditions except signalling:**
  - Test TC-CAF-APM-006: Test passed. Validated.
- **12.3.13 Define evaluated reaction depending on incident:**
  - Test TC-CAF-APM-027: Test passed. Validated.
  - Test TC-CAF-APM-028: Test passed. Validated.
  - Test TC-CAF-APM-029: Not evaluated. Although the first specification of the prototype included GoA level inhibition by JP and SP constraints. Finally, it was left out of scope as the initial defined procedure to recover from this situation is manual takeover (GoA1/2). Therefore, it was not evaluated.
- **12.3.14 Define reflexive reaction depending on incident:**
  - Test TC-CAF-APM-030_1: Test passed. Validated.
  - Test TC-CAF-APM-030_2: Test passed. Validated.
  - Test TC-CAF-APM-031: Test passed. Validated.
  - Test TC-CAF-APM-032: Test passed. Validated.

The following Table 1 and Table 2 contain the list of developed tests with the corresponding results obtained during their execution, separated into APM and ADM prototypes. A more detailed description of the tests, the traceability and validation evidence can be found in the following documents:

See D10.3_CAF_Developed_functions_APM_Annex_A.xlsx

See D10.3_CAF_Developed_functions_ADM_Annex_B.xlsx

Each test outcome is classified as follows:

- **Validated**: The test was successfully passed.
- **Not evaluated**: The test was not evaluated.
- **Fulfilment justified**: TRL6 tests that were not tested due to retrofit constraints, but compliance is demonstrated through reasoning.

Test ID	Description	Status TRL4 PC Environment	Status TRL5 Target Environment	Status TRL6 Demonstrator	Comments
TC-CAF-APM-001_1	Mission profile hosts two consecutive tasks that imply same movement direction. Mission profile is provided by C28 simulator and APM shall: 1. APM in GoA4 status 2. Enable one cabin 3. Engage ADM 4. Enable same cabin 5. Engage ADM	Validated	Validated	Fulfilment Justified	Running direction is not validable on demonstrator due to not upgradable ERTMS baseline
TC-CAF-APM-001_2	Mission profile hosts two consecutive tasks that imply opposite movement directions. Mission profile is provided by C28 simulator and APM shall: 1. APM in GoA4 status 2. Enable one cabin 3. Engage ADM 4. Enable opposite cabin 5. Engage ADM	Validated	Validated	Fulfilment Justified	Running direction is not validable on demonstrator due to not upgradable ERTMS baseline
TC-CAF-APM-002_1	Determine running direction when there is no required direction change on Mission Profile:	Validated	Validated	Fulfilment Justified	Running direction is not validable on demonstrator due to not upgradable ERTMS baseline

Interne

TC-CAF-APM-002_2	Determine running direction when there is required direction change on Mission Profile	Validated	Validated	Fulfilment Justified	Running direction is not validable on demonstrator due to not upgradable ERTMS baseline
TC-CAF-APM-003_1	REP sends a horn sound location in static segment profile. Mission profile providen on C28 has a horn sound location and C28 simulator provides that information to APM: 1. APM in GoA4 statys 2. APM shall enable cabin 3. APM shall engage ADM 4. APM shall command horn on C55 when C16 reports position stablished on C28 for horn	Validated	Validated	Fulfilment Justified	Disabled application on demonstrator due to noise restriction on populated areas
TC-CAF-APM-003_2	APM detects incident and command horn (maintenance workers detected by per) 1. APM in GoA4 status 2. PER sends maintenance workers detected on C8 3. APM demands horn on C55	Validated	Validated	Fulfilment Justified	Disabled application on demonstrator due to noise restriction on populated areas
TC-CAF-APM-004	APM in GoA4 status, TCMS on C55 sends a horn failure to APM and APM reports to REP on C28	Validated	Validated	Not Evaluated	Horn diagnostics not implemented on TCMS version

Interne

TC-CAF-APM-005	APM enters on GoA3 and sends driver activity control inhibition true on C55. Same request for GoA4 state and false on other states. State transitions triggered by Mission profile with TCG on C28	Validated	Validated	Fulfilment Justified	SW Inhibition of Driver activity control not allowed for safety reasons on demonstrator because it is a separate HW device already existing before GoA4 upgrade. Inhibition is achieved by simulating periodic input trigger from APM through TCMS as driver should do.
TC-CAF-APM-006	Direction controller only in forward if departure authorized under self-defined authorization conditions: No trackside inhibition APM-5.5 1. APM in GoA4 state 2. REP sends trackside departure inhibition on C28 and APM does not engage ADM 3. Trackside departure inhibition is removed on C28 and APM command engaging ADM through C16	Validated	Validated	Validated	
TC-CAF-APM-007	APM starts on AV state and commands direction controller in neutral, under MP instruction gains GoA3/4 and departure is allowed then direction controller is set to forward	Validated	Validated	Validated	

TC-CAF-APM-008	APM gets state GoA4 and departure is not inhibited, then with JP available and ADM in ready APM commands engage on C16 to ATP	Validated	Validated	Fulfilment Justified	As TP/ETCS upgrade is not authorized and C16 inputs on TP are not taken into account, engagement process is done through C55 + SS139
TC-CAF-APM-009	REP sends an ES request on C28 to APM and APM if GoA3/4 request ES to ETCS on C16	Validated	Validated	Fulfilment Justified	As TP/ETCS upgrade is not authorized and C16 inputs on TP are not considered, Emergency Stop is simulated by requesting full service brake to TCMS on interface Adaptors-TCMS
TC-CAF-APM-010	Passive coupling scheduled on mission profile only allowed if HB applied	Validated	Validated	Validated	
TC-CAF-APM-011	Active coupling, Mission profile has Active coupling and REP inhibition is deactivated	Validated	Validated	Validated	
TC-CAF-APM-012	APM request entering cuopling mode when active coupling and speed is under coupling max speed config(1/2km/h)	Validated	Validated	Validated	
TC-CAF-APM-013	Monitor coupler status on passive/active coupling and report to REP, read from C55 and send on C28. Simple situation where current task is TCP	Validated	Validated	Validated	
TC-CAF-APM-014	MP with TUC, ut status is coupled and LOZ reports standstill, APM starts	Validated	Validated	Validated	

	splitting and commands splitting on C55				
TC-CAF-APM-015	Read the splitting status of the train from the C55 interface	Validated	Validated	Validated	
TC-CAF-APM-016	Check that if APM is leading executes some functions and if it is non-leading does not	Validated	Validated	Not Evaluated	Only one unit was retrofitted on TRL6, so interaction between two units with APM could not be tested
TC-CAF-APM-017	APM executes a coupling task followed by a splitting task, before ending coupling task virtual cab must be deactivated	Validated	Validated	Validated	
TC-CAF-APM-018	APM in GoA4 and TCMS reports on C55 driver activated-cabin, report on C28. APM transition to GoA2 and then cabin is activated by APM, report anomaly on C28 status report train_anomaly 14, cab anomaly	Validated	Validated	Validated	
TC-CAF-APM-019_1	Mission profile is updated when train in mission but standstill, APM modifies mission including current task	Validated	Validated	Not Evaluated	The mission profile update was intentionally not implemented in the TRL 6 version due to safety considerations, as its inclusion could lead to both intended and unintended movements outside controlled scenarios.
TC-CAF-APM-019_2	Mission profile is updated when train in mission and moving, mission profile is	Validated	Validated	Not Evaluated	The mission profile update was intentionally not implemented in the TRL 6 version due to

	modified starting from next task				safety considerations, as its inclusion could lead to both intended and unintended movements outside controlled scenarios.
TC-CAF-APM-020	APM gets mission profile and execute tasks reporting to REP	Validated	Validated	Validated	
TC-CAF-APM-021	When TCG task changes APM GoA level, it shall start doing the Start of Mission	Validated	Validated	Fulfilment Justified	As TP/ETCS upgrade is not authorized there is no implementation of automatic data entry on demonstrator
TC-CAF-APM-022	APM checks that there are no passengers on the train with the data received from REP on the mission profile before starting TCL task	Validated	Validated	Not Evaluated	No scenario with passengers was allowed by the legal test regulation. Out of scope in TRL6.
TC-CAF-APM-023	APM is in GoA1 drive and Mission profile task status are reported when when tasks completed	Validated	Validated	Validated	
TC-CAF-APM-024	APM executes GoA4 task and TCG is next, APM waits until driver takes cabin	Validated	Validated	Not Evaluated	No scenario for testing TCG was defined. As it was not a priority scenario.
TC-CAF-APM-025	APM executes TJP and TCG task simultaneously from GoA4 to GoA2 and then in reverse	Validated	Validated	Not Evaluated	No scenario for testing TCG was defined. As it was not a priority scenario.

Interne

TC-CAF-APM-026_1	APM executes MP and last task is TSH, when tcms is in shutdown apm terminates mission,then reports to REP	Validated	Validated	Fulfilment Justified	On TRL6 when APM is executing TSH, power is removed before TCMS entering shutdown mode and before finishing the task so it is not possible to report it as APM loses power. This could be addressed, connecting APM's power supply to the train's main battery unit. This requirement is established for the Wake-up and Shut-down on D6.5. This was not the case for our prototype in the Demonstrator, as it was connected to the general power supply.
TC-CAF-APM-026_2	APM executes MP in GoA1/2 and last MP task is finished, checks mission termination	Validated	Validated	Validated	
TC-CAF-APM-027	APM gets objects detected by PER and evaluates reaction and reports to REP	Validated	Validated	Validated	
TC-CAF-APM-028	APM ends mission and request RAP by setting dir ctrl to neutral	Validated	Validated	Validated	
TC-CAF-APM-029	APM has a TJP on GoA3/4 where after one station GoA3/4 is inhibited, APM reports and wait	Not Evaluated	Not Evaluated	Not Evaluated	Although the first specification of the prototype included GoA level inhibition by JP and SP constraints. Finally it

					was left out of scope as the initial defined procedure to recover from this situation is manual takeover (GoA1/2). Therefore, it was not evaluated.
TC-CAF-APM-030_1	APM reacts to obstacle detected by per in the train gauge sending an EoA to ETCS, ADM brakes before obstacle with ETCS braking curve	Validated	Validated	Fulfilment Justified	As TP/ETCS upgrade is not authorized there is no implementation of EOA handling by ETCS it is managed by the Adapter with ETCS on FS or STM.
TC-CAF-APM-030_2	APM reacts to obstacle detected by PER in the train gauge sending an EoA to ETCS and removing EoA when obstacle disappears. ADM is not able to brake within ETCS braking curve and ETCS intervenes. APM waits after stopped for ISM action.	Validated	Validated	Fulfilment Justified	As TP/ETCS upgrade is not authorized there is no implementation of EOA handling by ETCS it is managed by the Adapter with ETCS on FS or STM.
TC-CAF-APM-031	APM reacts to obstacle detected by per in the train gauge sending an EoA to ETCS, ADM brakes before obstacle with ETCS braking curve, after stopped RAP is requested by setting dir ctrl to neutral	Validated	Validated	Validated	
TC-CAF-APM-032	APM has TCP on mission profile, Train is detected by PER and distance reported to APM, APM sends the information to ETCS until coupled	Validated	Validated	Validated	

Table 1: CAF APM validation tests

Figure 28: CAF APM test results

Some ADM requirements that are specified in the document have been already tested and validated on TSI2023 ATO commercial baseline. Consequently, the current validation scope focuses exclusively on the GoA3/4 delta:

2.3.2.6.2 ADM Functions:

- **12.2.1 Control initial traction effort**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.2 Determine maximum authorized speed**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.3 Respect JP Timing Points and Optimize the consumption**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-001: Test passed. Validated.
- **12.2.4 Stop exactly at the same intended location**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.5 Compute Operational Speed and Acceleration**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.6 Acquire train and ADM data**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-002: Test passed. Validated.
- **12.2.7 Regulate traction and braking**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-003: Test passed. Validated.
- **12.2.8 Supervise service brake efficiency during operation**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.9 Request holding brake**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.11 Check departure conditions**

- SS125 Requirements, already validated in GoA2 2023 TSI.

- **12.2.12 Detect that JP has been completed**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
- **12.2.13 Determine ADM state**
  - SS125 Requirements, already validated in GoA2 2023 TSI.
  - TC-CAF-ADM-005: Test passed. Validated.
- **12.2.14 Manage low adhesion**
  - SS125 Requirements, already validated in GoA2 2023 TSI.

Test ID	Description	Status TRL4 PC Environment	Status TRL5 Target Environment	Status TRL6 Demonstrator	Comments
TC-CAF-ADM-001	When a JP is received, if the train is stopped within the range of a TP it marks as reached, otherwise selects the next TP.	Validated	Validated	Validated	
TC-CAF-ADM-002	Train data is acquired by ADM (not safety): - Check TCMS train data - Check REP train data.	Validated	Validated	Validated	
TC-CAF-ADM-003	ATO-OB does not move the train and HB is requested when virtual cab is activated until departure conditions are fulfilled.	Validated	Validated	Validated	
TC-CAF-ADM-004	Test the JP applicable conditions for ATO operational.	Validated	Validated	Validated	
TC-CAF-ADM-005	Check behavior of ATO under different ETCS states.	Validated	Validated	Fulfilment Justified	As TP/ETCS upgrade is not authorized therefore AD mode is not implemented and it is not allowed to drive ADM on OS mode.

Interne

SS125	The SS125 requirement tests are already validated on TSI2023 tests	Validated	Validated	Validated	

Table 2: CAF ADM validation tests

Figure 29: CAF ADM test results

After executing all the defined test scenarios, it can be confirmed that most of the implemented functions for each prototype have been validated. The results demonstrate compliance with the functions and requirements specified in the T6.5 deliverable GoA34_R2D_1.1.1 document for both of our prototypes: Automatic Processing Module (APM) and Automatic Driving Module (ADM).

The validation process has been carried out across multiple configurations, such as, Local, target and train, where the last one has been demonstrated in the WP38 Demonstrator and was focused on the integration of the system. The process has provided evidence of correct system behavior under simulated and near real operation conditions.

2.3.3 Feedback for Specification and/or Demonstrators

There are some points that can be improved or clarified from CAF's perspective in order to get a more complete specification:

1. Coupling mechanism could require further analysis. APM currently takes its active/passive status from REP on C28 but that can lead to potential issues in case there was a mismatch on mission profiles on APM systems belonging different units in same consist (for example mission profile updated successfully on one unit and failed on another in same consist). The same mechanism as ETCS does would be recommended to get that information about train composition from TCMS, either through C55 or through dedicated communication between both APM systems through coupler gateway.
2. When determining the train direction, APM needs a JP packet and ETCS information. The JP packet is received after a JPReq packet is sent to REP. According to CAF's understanding, this JP is sent to APM upon establishing a C22 (SS126) connection with REP, after which JP and SP updates are maintained.

   During implementation, several points on the mechanisms were not fully defined or definition was too scattered, recommendation is to dedicate more detailed information about the process in the requirements.
3. Based on the experience gained during WP38 regarding the Demonstrator, it is important to correctly adjust the TJP ending location and TCP starting locations. If a passive train (The

one to be coupled) is stopped before the TJP ending location of a moving train (due to data issue or human error), the moving train will stop in front of the other train (at the EOA), and the coupling task could not be initiated as the TJP is considered incomplete.

Minor errors were also detected in WP6 T6.5 documents:

ID	Document	Section	Comment
1	LA_GoA34_R2D_1.1.1_Appendix_A	packet: PER_APM_OBSTACLE_DETECTIONS variable: NC_OBST_CLASS	Change variable type from Uint16 to Uint8. Currently there are 6 possible values for the variable. Uint8 ensures 255 different values.
2	LA_GoA34_R2D_1.1.1	12.3.6	Change 1m/s to 1-2km/h to maintain consistency with the document. As in other parts it is as 1-2km/h.
3	LA_GoA34_R2D_1.1.1	12.2.13	(see chapter 8.4.2) change to 8.6.2
4	LA_GoA34_R2D_1.1.1	ADM-10.1	It should be a TCMS requirement.
5	LA_GoA34_R2D_1.1.1_Appendix_A	packet: ETCS_APM_AVAILABLEFUNC variable: Q_ACK_POSTTRIP_EX_DIST_AVAIL	Values column is a copy paste of the variable above (Q_ACK_STANDSTILLSUP_AVAIL) it shall be corrected
6	LA_GoA34_R2D_1.1.1_Appendix_A	packet: NAV_FROM_LZ variable: M_TRAIN_HEADING	M_TRAIN_HEADING is an int16 but valid range described in values column exceed int16 range "0 .... 35999" and also unknown value, it shall be corrected, or variable type shall be changed to uint16
7	LA_GoA34_R2D_1.1.1_Appendix_A	packet: PER_APM_VISUAL_EVENT_DETECTION variable: NC_VISUAL_EVENT_CLASS_ATTRIBUTE	Description of enum: 41 = crossing train + SAL (Signal d'Alerte Lumineux) shall be passed to English or is it something specific for France?

Table 3: CAF Errors and feedback detected in the specification

Interne

2.4 FS

2.4.1 Methodology

The FS development and test activities follow the phases suggested by the EN 50761 [1] and the standard V cycle model reported in Figure 30.

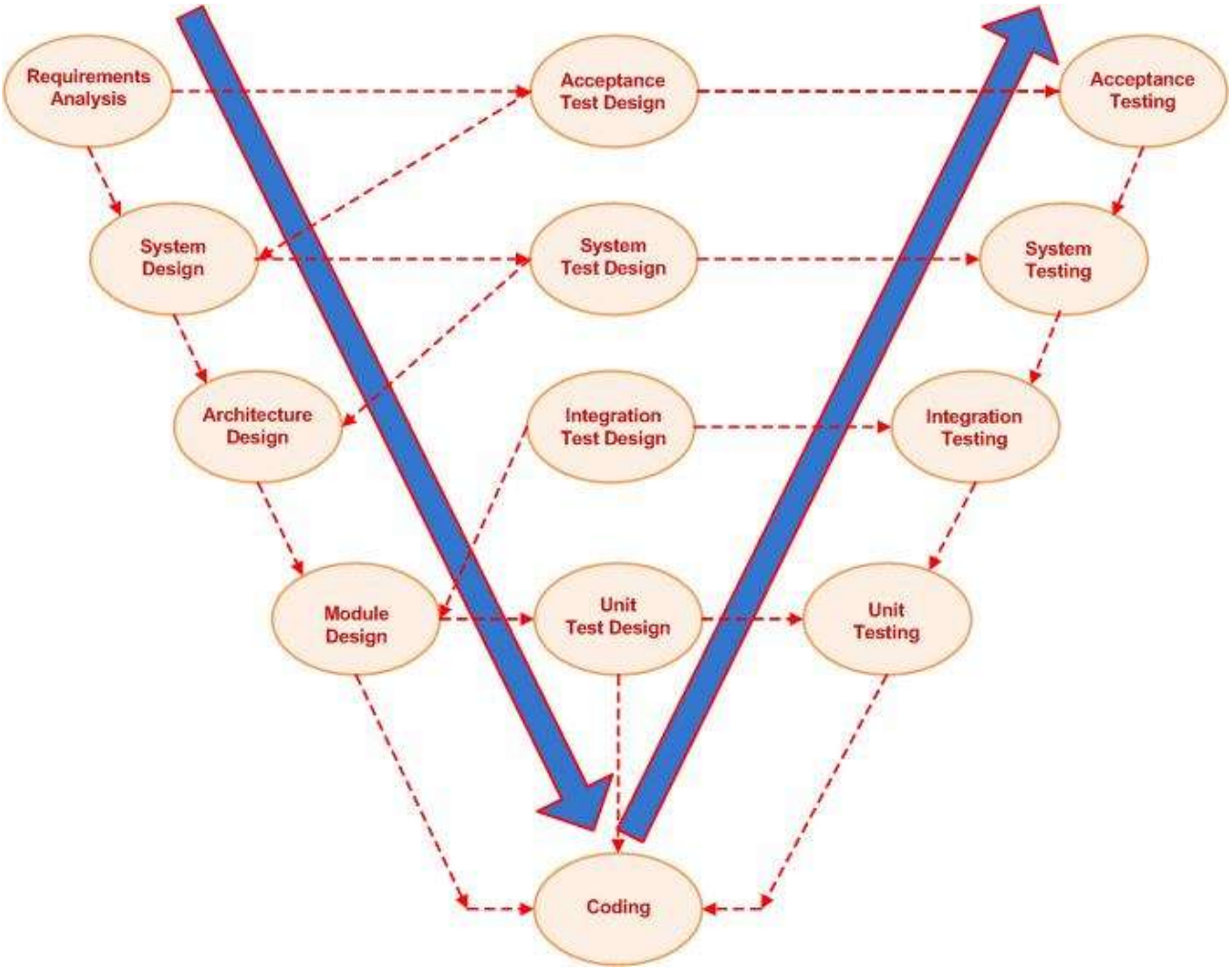

Figure 30: FS V Cycle Model

In particular, the development of the ADM subsystem starts from the definition of the architecture through the components and related interfaces, according to the Subsystems Requirements Specification [1]. In the development process, a model-based software development platform is then selected. The software implementation of the various components of the system and their integration are reported in the Design Description [6] via model block diagram.

The development activities take advantage of the Capella model available from the set of specifications [10] to clarify ambiguities in the ADM textual requirements, also providing the opportunity to analyse the related functional exchanges and sequence diagrams.

As a further activity, the test environment is conceived to appropriately stimulate the interfaces and behaviours of the developed subsystem and to allow the consequent testing activity to be executed and reported in the Test Report [8], according to the Test Specification [7].

Interne

2.4.2 Test Activities Description

The FS developed prototype demonstrates ATO over ETCS System function according to system architecture specification described to D6.5 [10], considering the Shift2Rail results.

According to WP6 level, in Figure 31 is reported, for each interface within Control Command and Signalling (CCS) on-board architecture, the basic target level of the information, equivalent to Functional Interface Specification (FIS).

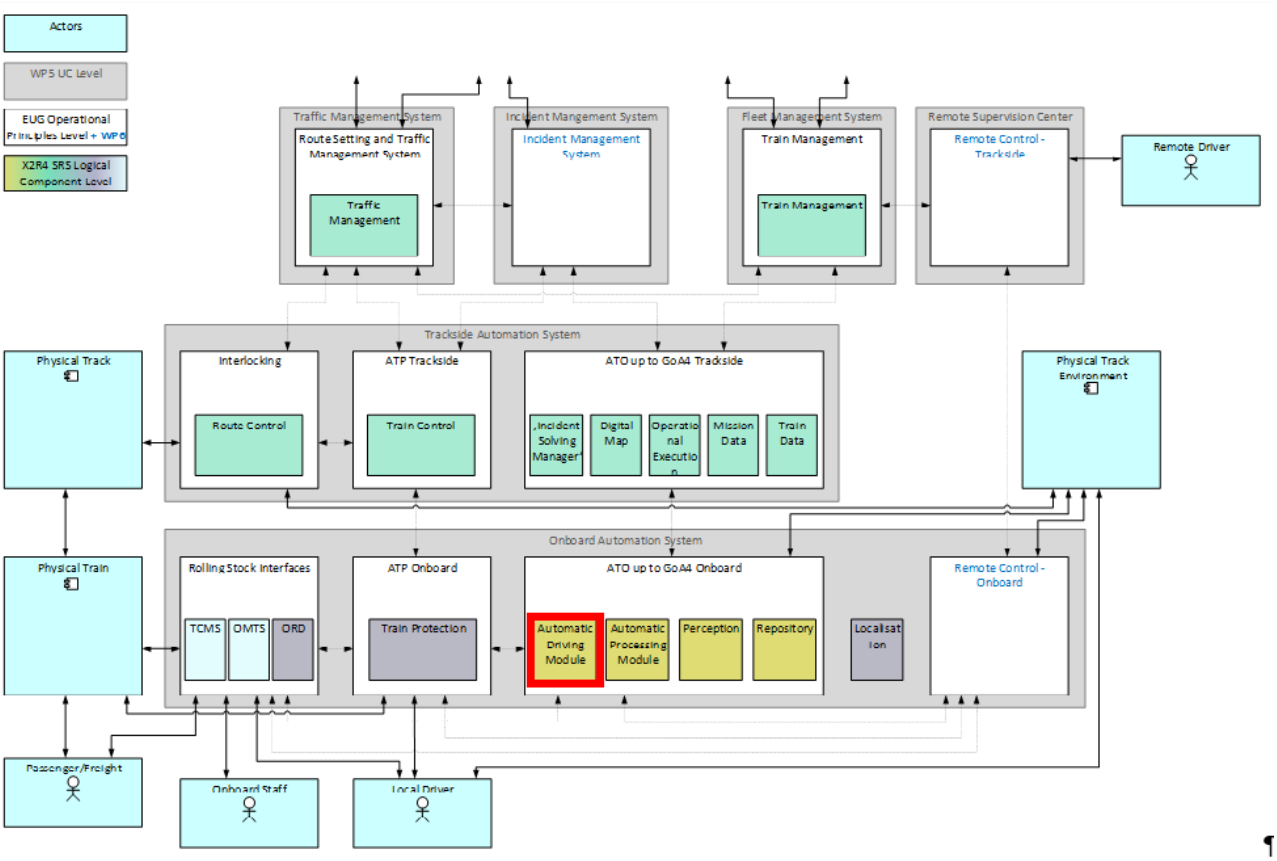

Figure 31: FS Focus on Automatic Driving Module (red box)

In Onboard Automation System the ATO up to GoA4 Onboard component is composed by:

- Automatic Driving Module: this on-board component has the responsibility to automatically drive a train.
- Automatic Processing Module: this on-board component should substitute driver and train attendant responsibilities for reacting in case of incident. It manages mission execution, safe reflexive reactions, evaluated reactions and safety procedures.
- Perception: this on-board component senses the Physical Railway Environment in place of a driver
- Repository: this on-board component communicates with trackside to acquire Journey Profile and infrastructure data from trackside, collects and reports incidents to trackside.

FS prototype is focused on ADM on board component, the red box Figure 31.

Interne

The scope of the test environment is limited to the testing of a subset of selected use cases and functions of the ADM as described below. Thus, a variety of functions provided by other ATO components, such as APM, ATP, Repository, Perception and Localization are not in the scope of the test activities. Consequently, the following interfaces of ADM are simulated to stimulate the functions under test (see Figure 32):

a) Train Protection, according to SS130: From/To Train Protection To/From ADM

b) Train Control and Monitoring System, according to SS139: From/To Train Control and Monitoring System To/From ADM.

c) Repository, according to C22: From/To Repository To/From ADM.

d) Localization, according to C9: From Localization To ADM.

Within this test environment, dedicated functions that require active interaction with systems external to the ADM have been developed in accordance with the specifications and have been implemented to appropriately trigger the functionalities developed, allowing verification of their correct behavior and implementation. Some of these functions are implemented as stubs, whose purpose is to provide the required input data without performing any real processing.

Moreover, additional interfaces, such as ATO_Commands and Diag, are included specifically for test purposes. These interfaces provide dedicated messages to trigger ADM scenarios and allow the collection and observation of internal variables that are not exposed through the interfaces defined in D6.5 [6]. This approach enables a more detailed verification of the component behavior and supports thorough validation of internal states and execution flows during testing.

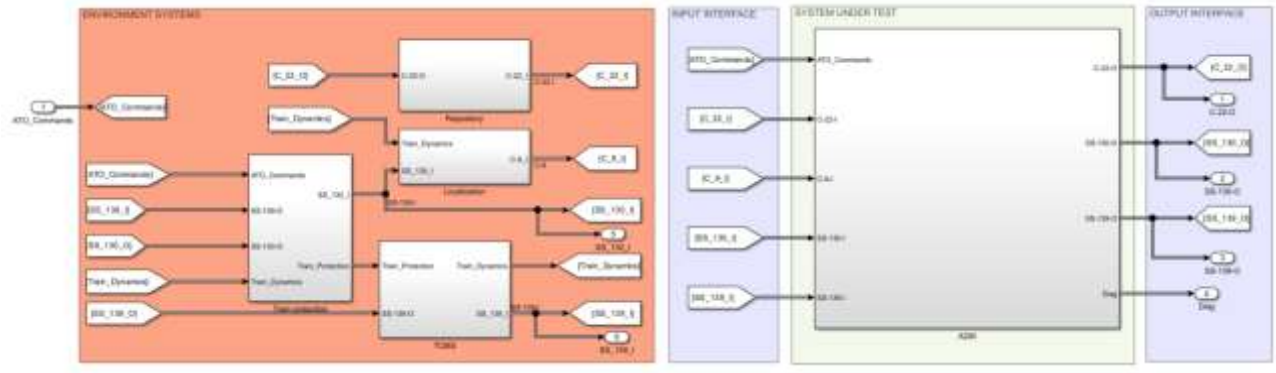

Figure 32: FS Test Environment

The dynamic model of a railway vehicle has been encapsulated into the TCMS to simulate its motion dynamics, ensuring the proper stimulation of the odometry data.

The GUI interface (Figure 33) provides a user-friendly layer that enables interaction with the simulation environment. Through this interface, users can manage the driver-related data inputs needed to initiate and reproduce simulations. The GUI also supports the monitoring of ongoing simulations and visualization of relevant train variables. This ensures that users can operate the environment without requiring direct access to the underlying code or data structures.

Together, ADM and GUI interface components establish a modular and extensible architecture, where the ADM environment ensures compliance with the functional and technical requirements, while the GUI interface enhances usability and accessibility for end-users.

Interne

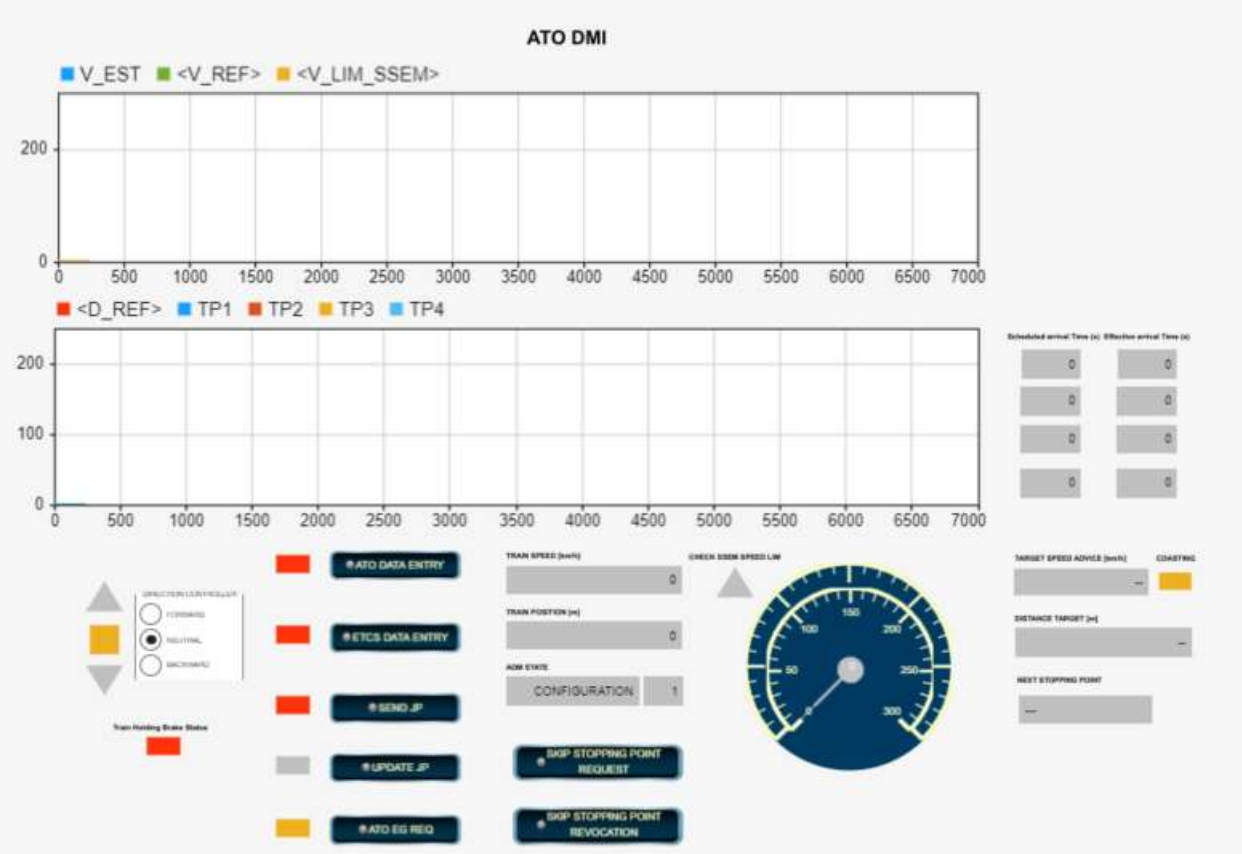

Figure 33: FS GUI for plots and commands

Subsequently, the testing phase is planned, aimed at confirming what has been developed with possible management of unexpected behaviour (bug-fixing) recorded in the Test Report [8].

The track layout configured in the simulation environment is the Bologna–Milano high-speed railway. This line is characterized by high-speed operation with a maximum velocity of 300 Km/h (see Figure 34), gradients and curve radii as shown in Figure 35. The simulation considers 202.690 km of track and includes 2 stations (located respectively in Bologna and Milano), with gradient and curvature data directly extracted from the RFI infrastructure data of the Bologna–Milano line.

Interne

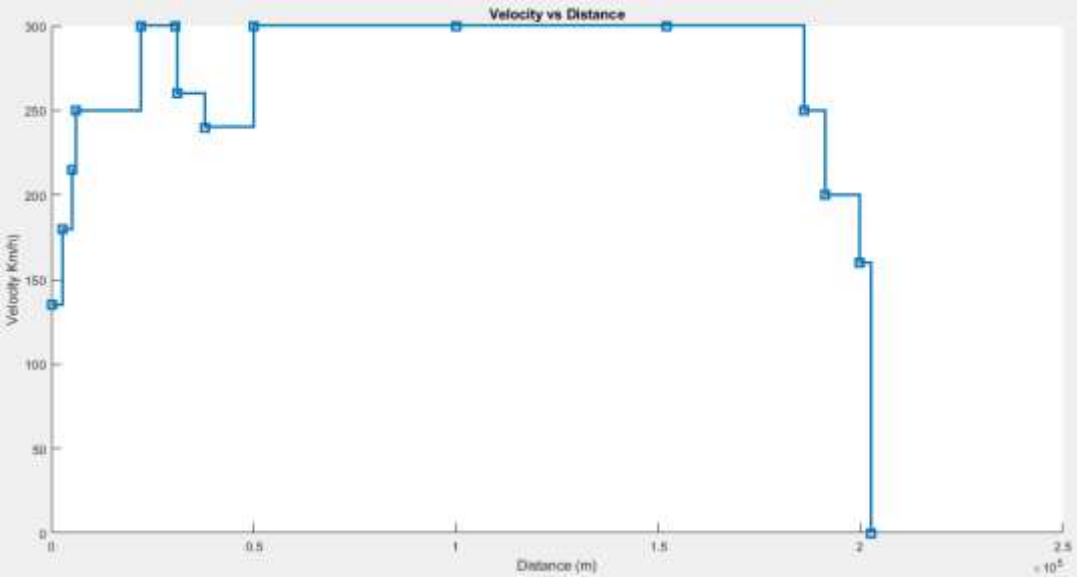

Figure 34: FS Track speed limits

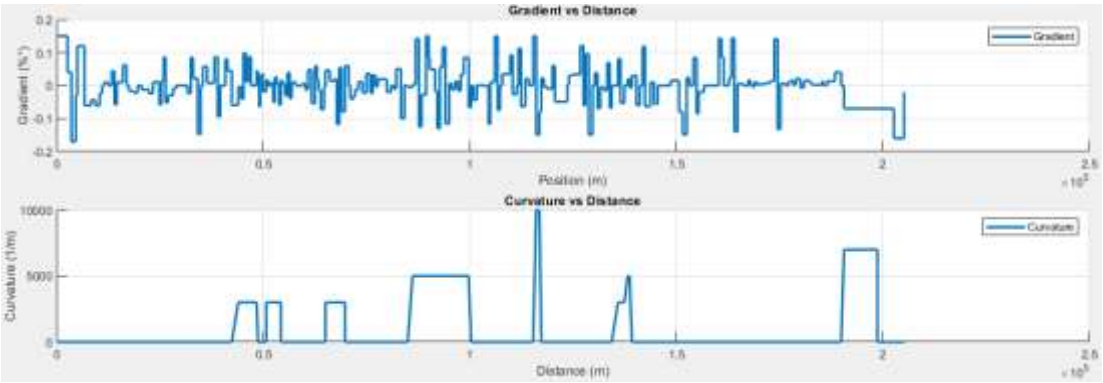

Figure 35: FS Gradient and curvature for the line

The profile defines Bologna as the starting point and Milano as the destination. The JP is composed of four Timing Points in which three are classified as Passing Points, while the fourth corresponds to the End of Journey, represented by Milano station. The expected completion time for the Journey Profile is 2,845 seconds, corresponding to approximately 47 minutes and 25 seconds. This duration represents the nominal reference time for the execution of the journey.

However, in the ADM implementation, the Journey Profile timing is not managed with second-level strictness. Instead, it allows for a certain degree of tolerance, so that minor deviations in arrival times can be accommodated without violating the overall operational plan.

Within this framework, a dedicated optimization function is designed in the trajectory planning process. This optimization simultaneously considers punctuality and energy consumption, enabling a balanced trade-off between adherence to the planned timetable and efficient use of traction energy.

As a consequence of this optimization, controlled variations in arrival times are expected. The system may anticipate higher traction effort on specific track segments and compensate on others requiring lower effort, intentionally exploiting small time margins to recover or redistribute time along the journey.

Such flexibility is essential to enable the optimization of multiple variables under real operational conditions. By allowing limited temporal adjustments, the system can improve overall performance,

Interne

such as energy efficiency and robustness to disturbances, while remaining consistent with the timing constraints defined by the Journey Profile and maintaining passenger comfort through fixed parametric limits on maximum jerk.

The characteristic parameters of the Journey Profile used for the execution of the use cases are listed in the following table.

BASIC_PASSENGER_JP Data				
Timing Point ID	1	2	3	4
Timing Point Position [m]	30.567	100.101	152.000	202.690
Arrival Time [s]	520	1.420	2.060	2.845
TP Info	Passing Point	Passing Point	Passing Point	End of Journey

Table 4: FS Basic Passenger JP Data

In UCs where the train type is not explicitly specified, a passenger train has been assumed as the reference configuration.

For freight train scenarios, the Journey Profile defined below is used as the reference for test execution and validation.

BASIC_FREIGHT_JP Data				
Timing Point ID	1	2	3	4
Timing Point Position [m]	30.567	100.101	152.000	202.690
Arrival Time [s]	740	2.300	3.480	4.672
TP Info	Passing Point	Passing Point	Passing Point	End of Journey

Table 5: FS Basic Freight JP Data

The train model used is derived from longitudinal train dynamics. In this formulation, the train is represented as a point mass with a single degree of freedom, subject to the effects of the traction/braking system, rolling and bearing resistances, aerodynamic drag, tunnel effects, as well as grade and curvature resistances (the present model follows the methodological principles outlined in [14]). The equations of motion are expressed as:

$$\dot{s} = v$$

$$M\dot{v} = -Av^2 - Bv - C - F_e(s) + u$$

Here, v (cm/s) and s (cm) denote the train's velocity and position, respectively. The term u (N) represents the commanded traction or braking force. The term F_e (N) accounts for external resistances originating from track conditions. The parameter τ defines the inertial delay in the longitudinal force response. The train mass is denoted by M (kg). The coefficient C (N) aggregates rolling and bearing resistances; B (Ns/cm) relates to aerodynamic input; A (Ns²/cm²) is the aerodynamic drag coefficient. The values of M, A, B, and C are dependent on the specific train characteristics. A starting resistance associated with dry friction is included in A as an additional

contribution. This effect increases resistance at low speeds, diminishing as velocity rises. In the reported dynamical model, the dependency by the time is omitted.

The external resistance F_e is the sum of two distinct contributions: gravitational resistance (F_g) and curving resistance (F_R), expressed as:

$$F_e(s) = F_g(s) + F_R(s)$$

$$F_g(s) = -Mg \cdot slope(s)$$

$$F_R(s) = -M \cdot 6 \times 10^6 / R(s)$$

where F_g(N) is the component due to track gradient, with slope (mm/cm) representing the track grade and g (m/s²) the gravitational acceleration. The term F_R (N) quantifies the resistance from track curvature, with R (cm) denoting the curve radius.

The train parameters configured in the test environment are based on the data reported in the section *8.2.1.3 Train Parameters* of the public deliverable [15], when applicable to the vehicle dynamics here described.

The test activities are designed to test the Automatic Driving Module (ADM) following functions selected by FS, whose requirements are provided by the WP6, in the deliverable D6.5:

1. Determine maximum authorized speed
2. Respect JP Timing Points and Optimize the consumption
3. Stop exactly at the intended location
4. Compute Operational Speed and Acceleration
5. Request holding brake
6. Determine ADM state

The simulation environment shall execute, at least, the following list of use cases to properly stimulate the selected ADM functions, considering a set of configuration data detailed below:

UC ID 1 - Stop at next station or rescue point

UC ID 2 - Traction and brake control

UC ID 3 - Request holding brake

UC ID 4 - Check departure conditions except signalling

UC ID 5 - Determine stopping point for freight or passenger train

For each use case, whose detail is reported in [10], the corresponding sequence diagram is reported in Figure 36, Figure 37, Figure 38, Figure 39 Figure 40, according to the input specification documents.

Moreover, the test plan places emphasis on a limited subset of requirements that were not adequately validated through the use cases above. These aspects are addressed with a more detailed verification strategy to ensure their proper implementation and compliance with the system specifications.

Interne

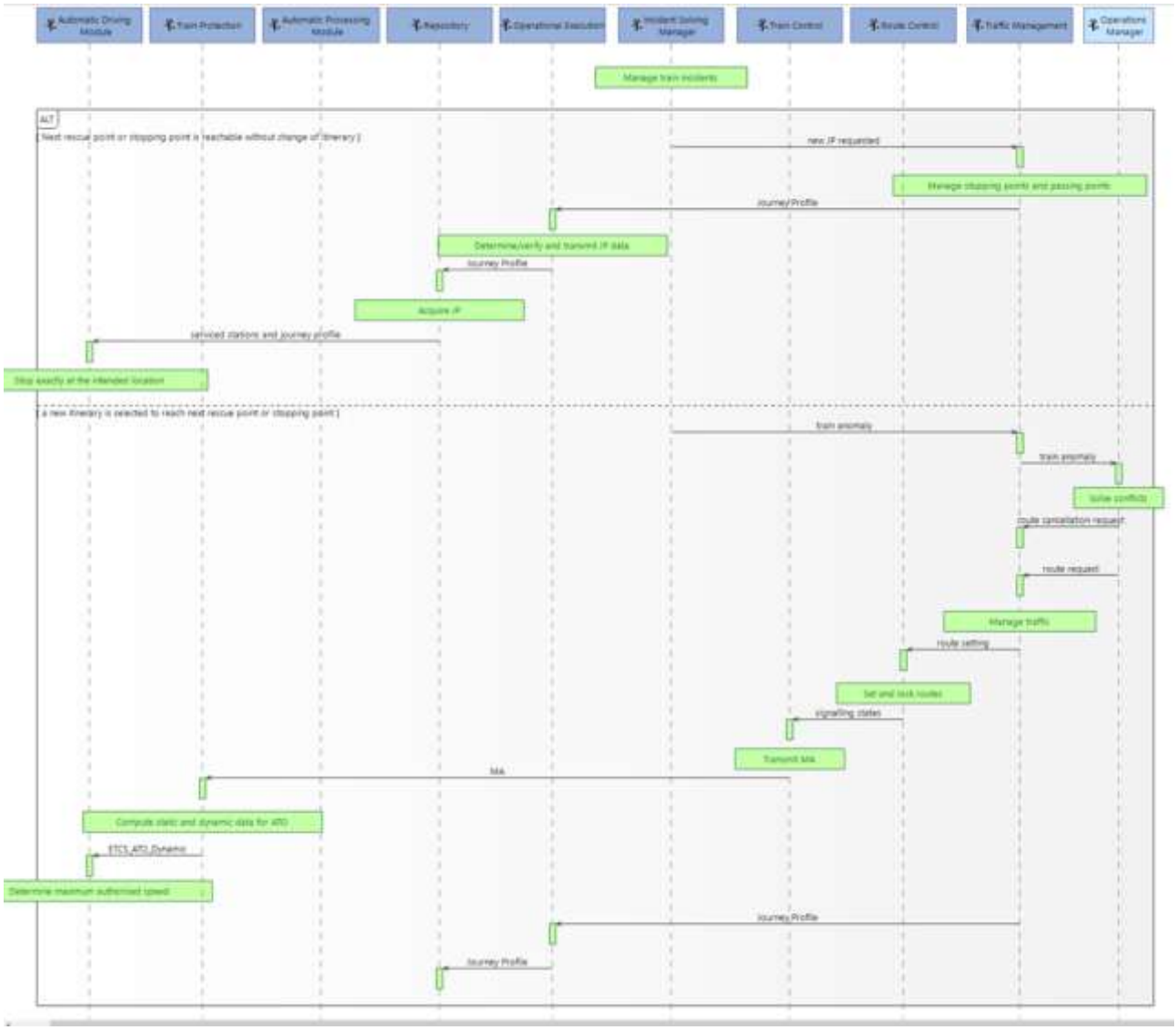

Figure 36: FS UC ID 1 sequence diagram

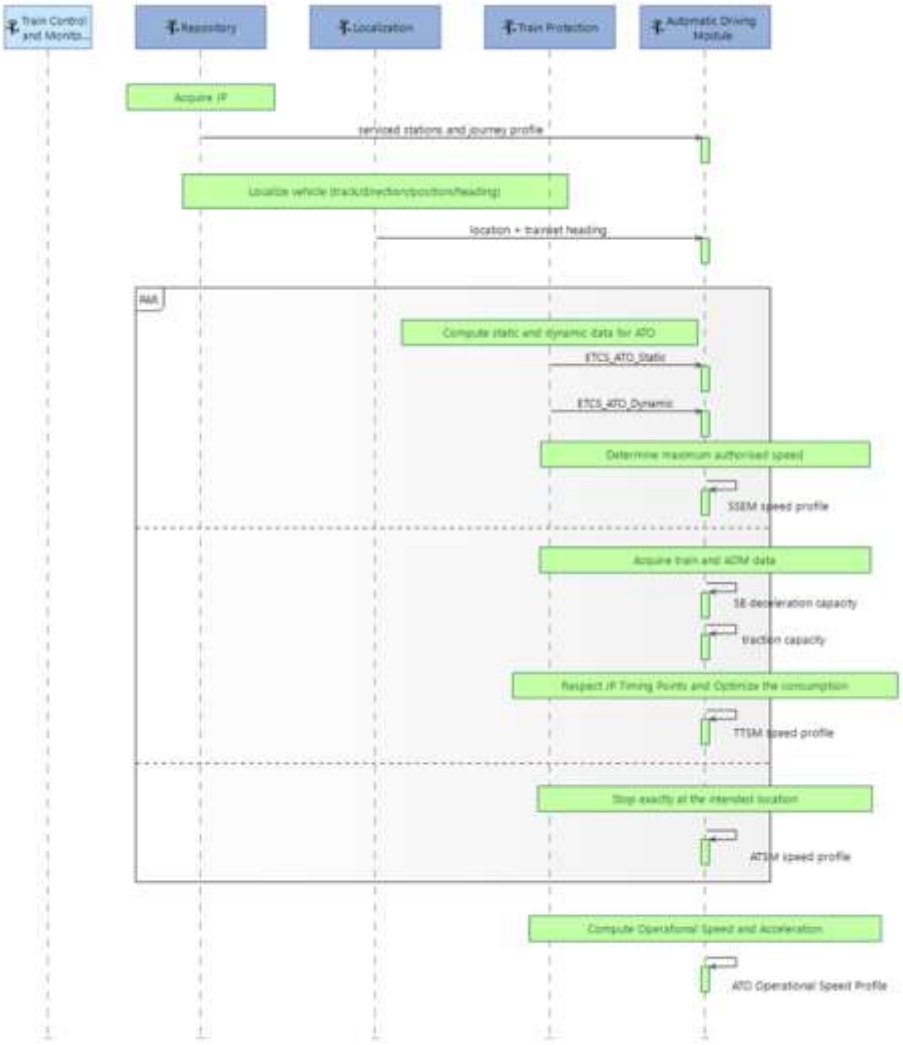

Figure 37: FS UC ID 2 sequence diagram

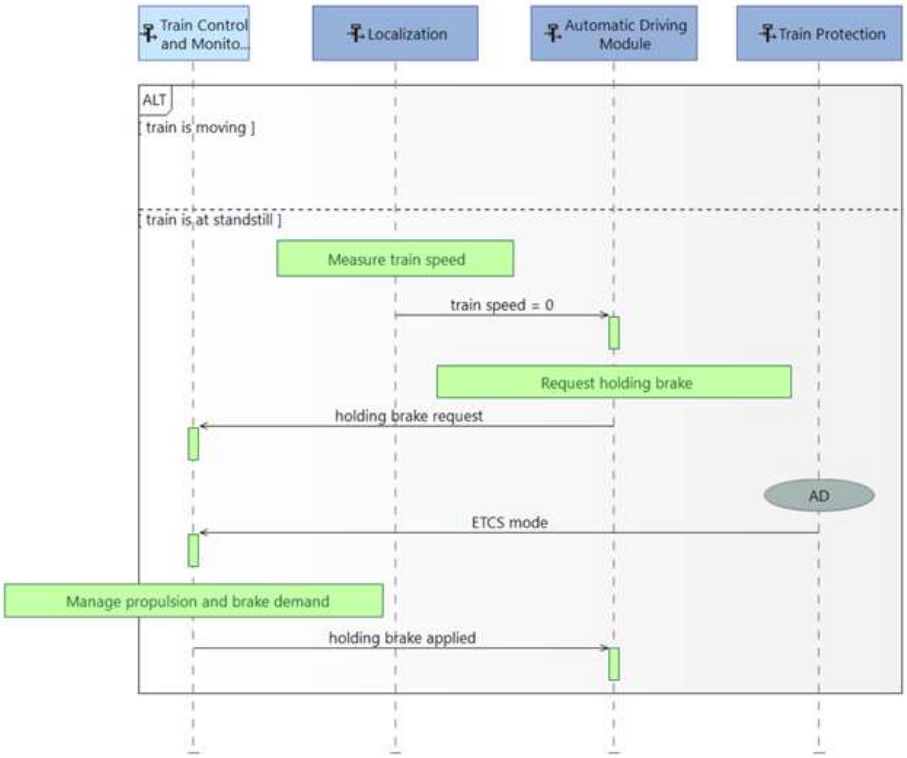

Figure 38: FS UC ID 3 sequence diagram

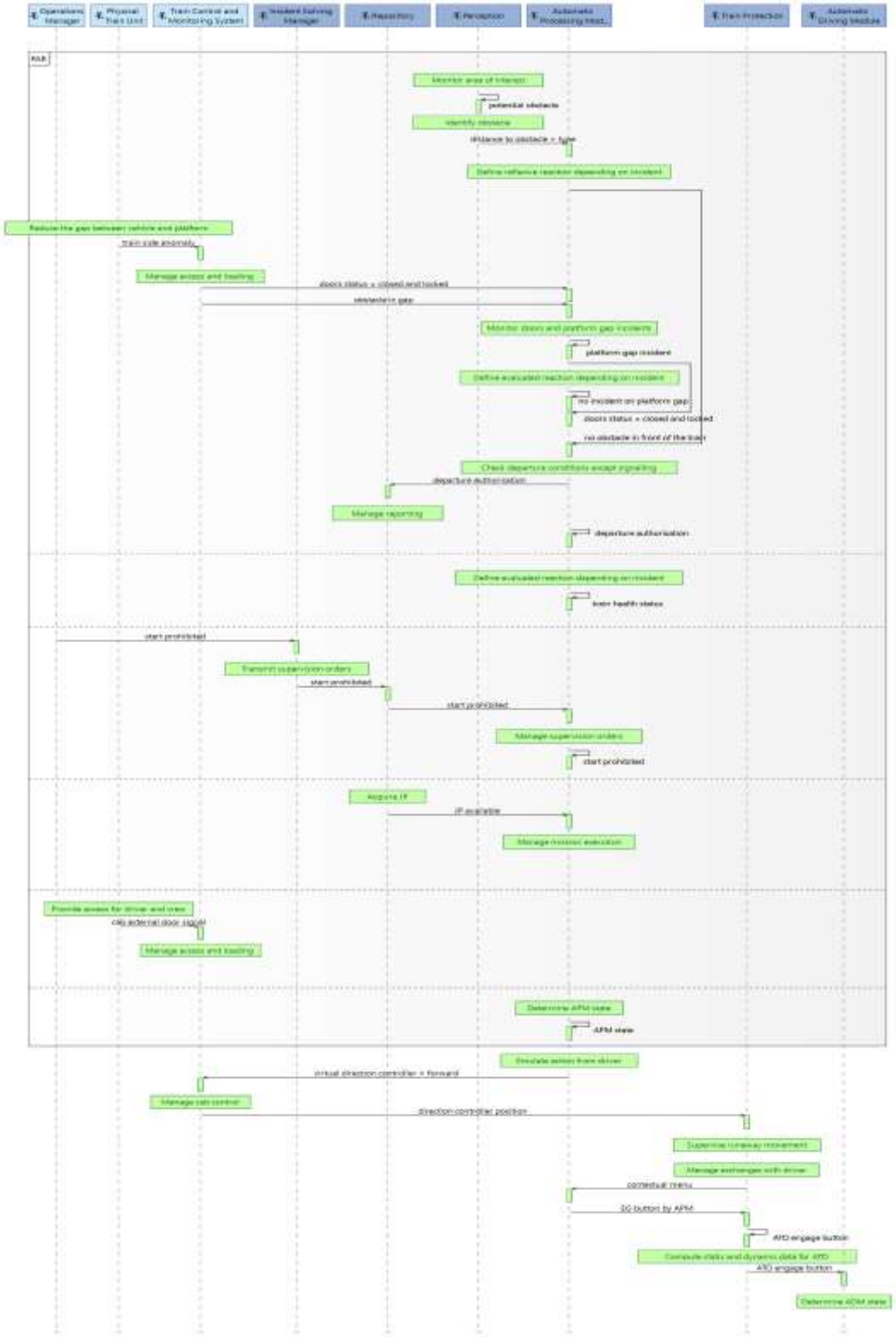

Figure 39: FS UC ID 4 sequence diagram

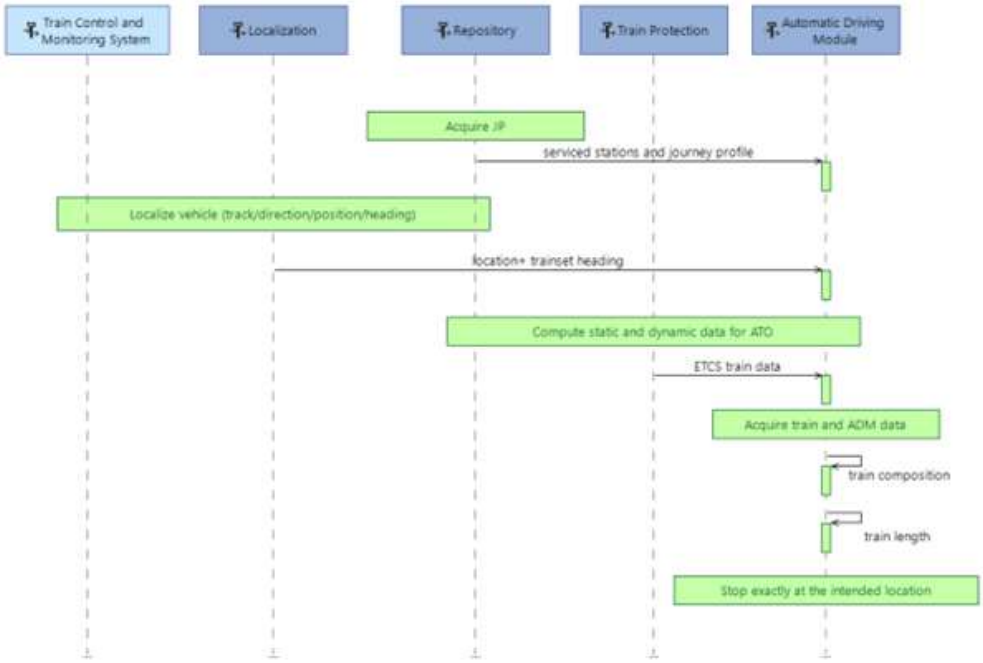

Figure 40: FS UC ID 5 sequence diagram

The following table indicates, by marking an 'X' in the corresponding column, in which use case each ADM function is executed, including the function reference number from [6] alongside its title.

ID - Function	UC ID 1 Stop at next station or rescue point	UC ID 2 Traction and brake control	UC ID 3 Request holding brake	UC ID 4 Check departure conditions except signalling	UC ID 5 Determine stopping point for freight or passenger train
12.2.2 - Determine maximum authorized speed	X	X			
12.2.3 - Respect JP Timing Points and Optimize the consumption	X	X			
12.2.4 - Stop exactly at the intended location	X	X			X

12.2.5 - Compute Operational Speed and Acceleration	X	X			
12.2.9 - Request holding brake			X		
12.2.13 - Determine ADM state				X	

Table 6: FS Use Cases and Functions Mapping

All the issues and the details on feedback are reported in the next section.

The following table outlines the status of the test execution activity.

Use Case ID	Use Case Description	Test ID	Test Executed	Test Result
1	Stop at next station or rescue point	1	x	PASSED
		2	x	PASSED
2	Traction and brake control	3	x	PASSED
3	Request holding brake	4	x	PASSED
4	Check departure conditions except signalling	3*	x	PASSED
5	Determine stopping point for freight or passenger train	5	x	PASSED
		6	x	PASSED

Table 7: FS Summary of Status of Use Cases Test Planning and Execution Activity

*Note: According to the specification of the use case "Check departure conditions except signalling" and to the corresponding sequence diagram, the only function of ADM which is activated by such use case is the ADM state machine. Such behaviour is stimulated by the Test ID 3.

As previously described, a limited subset of requirements that were not adequately validated through the use cases above have been tested in dedicated Test Cases:

Test ID	Requirements to test	Test Executed	Test Result
7	SS125 7.1.3.2	x	PASSED
8	SS125 7.1.3.3, SS125 7.1.3.7, SS125 7.1.3.8	x	PASSED
9	SS125 7.1.3.6	x	PASSED

10	SS125 6.2.1.8	x	PASSED
11	SS125 7.3.1.2	x	PASSED

Table 8: FS Summary of Status of Test Planning and Execution Activity

2.4.3 Feedback for Specification and/or Demonstrators

During the development, test specification and execution, some remarks have been collected, discussed with the WP10 partners and shown here. The following table describes the FS feedback for ADM specification resulting from the development and validation activities currently executed.

Chapter/Figure/Table	Comments (justification for change)	Proposed change
D6.5 – Annex 1- SRS ATO up to GoA3/4, 10.5.6 C22	"T_Departure_Seconds" is repeated. It is present "departure time" that is not clear, seems to repeat "T_Departure_Seconds"	Remove the duplicated "T_Departure_Seconds" term and "departure time".
D6.5 – Annex 1- SRS ATO up to GoA3/4, 10.5.6 C22	"T_Departure_Date" is missing to complete the information "T_Departure_Seconds"	Add "T_Departure_Date".
D6.5 – Annex 1- SRS ATO up to GoA3/4, 10.5.6 C22	"Current Limitation" is repeated.	Remove the duplicated "Current Limitation" term.
D6.5 – Annex 1- SRS ATO up to GoA3/4, 10.5.6 C22	"low adhesion reported by the driver" present into exchange items from ADM to REP, suggests that it is sent from board to trackside but for GoA4 it is assumed that there is no driver on board.	Clarify the description of the "low adhesion reported by the driver" for GoA4 application.
D6.5 – Annex 1- SRS ATO up to GoA3/4, 10.5.6 C22	In "serviced stations and Journey Profile" Journey Profile seems to already include information about serviced stations (Stopping Points infos).	Modify "serviced stations and Journey Profile" with "Journey Profile" or better detail how to represent serviced stations informations.
D6.5 – Annex 1- SRS ATO up to GoA3/4, 10.5.6 C22	"routing error" corresponds to bit1 of the Q_STR_Indicators of the Status Report packet and therefore should be associated with REP_ADM_Status_Report and not REP_ADM_Journey_Profile	Relate "routing error" to REP_ADM_Status_Report.

D6.5 – Annex 1 - LA_GoA34_R2D_1.1.1_Appendix_A_20250127	The "Packed Details" sheet doesn't include all interfaces related to H column in "Packet List" sheet.	Add the missing interfaces packet details.
D6.5 – Annex 1 - LA_GoA34_R2D_1.1.1_Appendix_A_20250127	The packet details do not include packets of SS139	Add the ADM-TCMS interface packet details.
D6.5 – Annex 1 - SRS ATO up to GoA3/4	The input/output tables report duplicated input/output terms	Remove duplicated terms
D6.5 – Annex 1 - SRS ATO up to GoA3/4, 13.7.12 Stop at next station or rescue point	In the UC “Stop at next station or rescue point” a rescue point is mentioned but in the JP packet there is no reference to “rescue point” data	Define “rescue point” information or remove it from the UC.

Table 9: FS Feedback for R2DATO WP6 Specifications

Moreover, the following feedbacks are considered applicable to Subset 125 and Subset 130, already approved by ERA, and are addressed by FS within the process of change requests to Subset 125 and Subset 130 out of the context of R2DATO project, according to the WP10 work group discussion.

Chapter/Figure/Table	**Comments (justification for change)**	**Proposed change**
12.2.2 Determine maximum authorised speed (SS125 7.1.3.2)	In ADM has to be evaluated the Speed and Distance monitoring mode since in Subset 130 RSM/TSM/CSM data is not provided but it is as M_SDMTYPE to JRU in SPEED AND DISTANCE MONITORING INFORMATION (SS027-Req. 4.2.4.20). Transmitting the data without duplicating the calculation in the ADM as in SS026 3.13.10.6.1 can reduce the system’s computational load and ensures that both ATO and ETCS systems consider the same mode, avoiding the risk of data misalignment and making ADM independent from ETCS configuration data.	Add SPEED AND DISTANCE MONITORING INFORMATION (SS027-4.2.4.20) to the interface between ETCS and ADM since it is missingin SS130.
12.2.2 Determine maximum authorised speed	In EBI computation, ADM needs parameters, such as T_bs, which are not available from SS130. To calculate T_bs, the system must have access to the configuration parameter T_brake_service, which defines the	Add SPEED AND DISTANCE MONITORING INFORMATION (SS027-4.2.4.20), SDM_BRAKE_AND_TRACTION_INTERFACE_CONFIGURATION (SS027-4.2.4.47) and TRAIN DATA (SS027-

Interne

(SS125 7.1.3.3)	braking characteristics of the train. To align the EBI/SBI computation between ATO and ETCS, there's the need to match ATO and ETCS configuration parameters which are not included into SS130. For example: in cases where T_bs1 and T_bs2 feedback values are not available, and the Service Brake (SB) is present, requirement 3.13.9.3.3.3 specifies that T_bs1 = T_bs2 = T_bs. If SB is not available, these values default to zero. All such values are considered configuration data. As per 3.13.6.3.2.4, T_bs = T_brake_service, a value that may vary depending on the special brake configurations currently in use, but still treated as a configured parameter rather than runtime data. Furthermore, requirement 3.13.9.3.6.2 states that, if feedback on the SB command is available, then the T_Indication value is to be calculated as 5 seconds + T_driver. Those information are present in SS027 to JRU, where it is available in the message SDM_BRAKE_AND_TRACTION_INTERFACE_CONFIGURATION (ref. 4.2.4.47, SS027) and TRAIN DATA (SS027-4.2.4.2) . Additionally, SUBSET 027 provides the parameters D_TARGET and V_TARGET, which represent the braking target distance and speed shown to the driver, as described in requirement 4.2.4.20.	4.2.4.2) to the interface between ETCS and ADM since they are missing in SS130.

Table 10: FS Change Requests for SS125 and SS130

Interne

2.5 GTSD

GTSD developed an APM component, as presented in [D10.1]. The methodology applied in the context of R2DATO WP10 is a tailoring of GTSD's system IVV processes (Integration, verification and validation) under the light of generic product innovation.

2.5.1 Methodology

The processes applied are:

1. System Component Testing, applied to APM
2. System Integration Testing, applied to the combination APM + PERception.

Component tests for PERception are a pre-requisite to 2), yet applied in the context of R2DATO WP11 within a development and testing environment dedicated to the PERception component.

The overall tailoring results in a process overview depicted in Figure 41, GTSD process for T10.3.

- Test definition

Each test definition takes as input a single use case in [27] or [28] and the associated requirements in [23] and, for system integration with PER, [24], as derived among others from [11]. One use-case can lead to several test cases. One test case originates from only one use case.

The test definition is articulated in following steps:

1. Specifying a test configuration: components under test and operational actors implied in the test.
2. Specifying initial test conditions: relations between actors that have to be fulfilled at test start, e.g. track topology, distance relationships between switches, curves, platforms, …train or human positions, or states or properties of precise actors (cab opened, switch position or level crossing barrier state, …). Keeping this step abstract allows later to vary several test configurations for one test implementation
3. Specifying the implementation: sequencing of actor actions/vehicle movements and the definition, along the sequence, of expected system or component behaviour/reaction (failure conditions). Definition of final success criteria.

Test cases are first specified for review by the developer.

- Review

During this step, the developers review the test cases specified. They issue questions, proposals for improvements, complaints. The process comprises the improvement of the test case specification.

- Test Configuration

The test configuration consists in defining, based on a test spec., the environment chosen for the test: railway track area hosting the test case implementation, rolling stock involved, human and non-human actors, track-side configuration e.g. track detection system and ISM. It also comprises the definition, for each actor, for each of these actors, the set of components that

Interne

implements it during the simulation: component(s) under test and/or simulator(s). Lastly, the wiring of these components is defined.

Concretely, it consists in setting under configuration management a track map, some train and user configuration and a configuration script. A test configuration can be shared by several implementations.

- Test Implementation

The test implementation consists in scripting the sequence the actions requested from the simulators and inputs fed to the components under test, that complete the test implementation. Along the test execution, the test report is scripted too. The test implementation comprises the scripting of the test failure or test success to the environment. Lastly, test logs may be appended to the report.

- Lab Integration

During lab integration, a release of the full test environment is assembled: a configuration of test environment, objects under tests, simulators and test implementations is chosen; It is then built on the official integration host, ready for test execution.

- Test Execution

The test execution is performed automatically with the test configuration chosen during lab integration. A test report is issued automatically (failed and passed tests) as well as detailed logs (component/simulator and script dependant).

- Test Evaluation

During the evaluation of a test session, the logs searched for test successes and fails. The root causes for failures are analysed. Test evaluation may require to perform dedicated tests again on developer hosts. It ends in some proposal(s) for improvement of components under tests, simulators or test scripts.

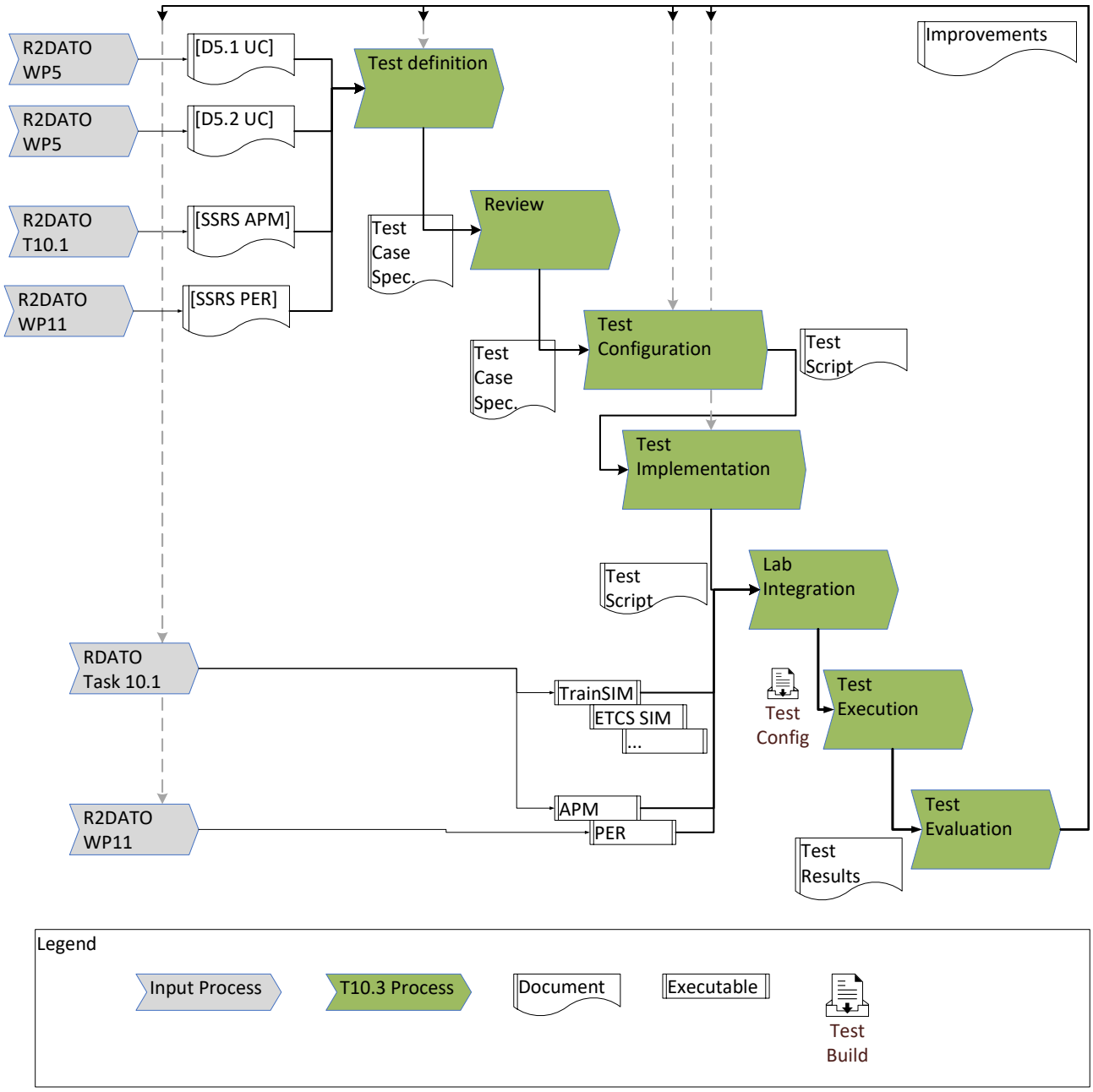

Figure 41: GTSD process

2.5.2 Test Activities Description

2.5.2.1 Equipment and Components Under Test

GTSD's standard vehicle environment is routinely deployed to test field products as depicted for Figure 42.

Interne

Figure 42: GTSD standard test environment for vehicle products

To test APM and its integration with PERception in the lab, this test environment was amended to

- support test scenarios implying human actors and fires, incl. management of video input if PER is under test.
- take advantage of the reduced need in hardware (videos required, rather than sensors boards as for ETCS or a link to some TCMS on a train network as for ADM)
- enable to scale back to system tests with ETCS & ATO in later iterations

It leads to server without I/O boards and enhanced video capabilities:

Figure 43: GTSD WP10-WP11 tayloring of GTSD's vehicle test environment

2.5.2.2 Components Under Test

The components under test are:

- APM: component test
- APM + PERception: system incremental testing, component tests for PERception happening in the context of R2DATO WP11

They are depicted in following figure:

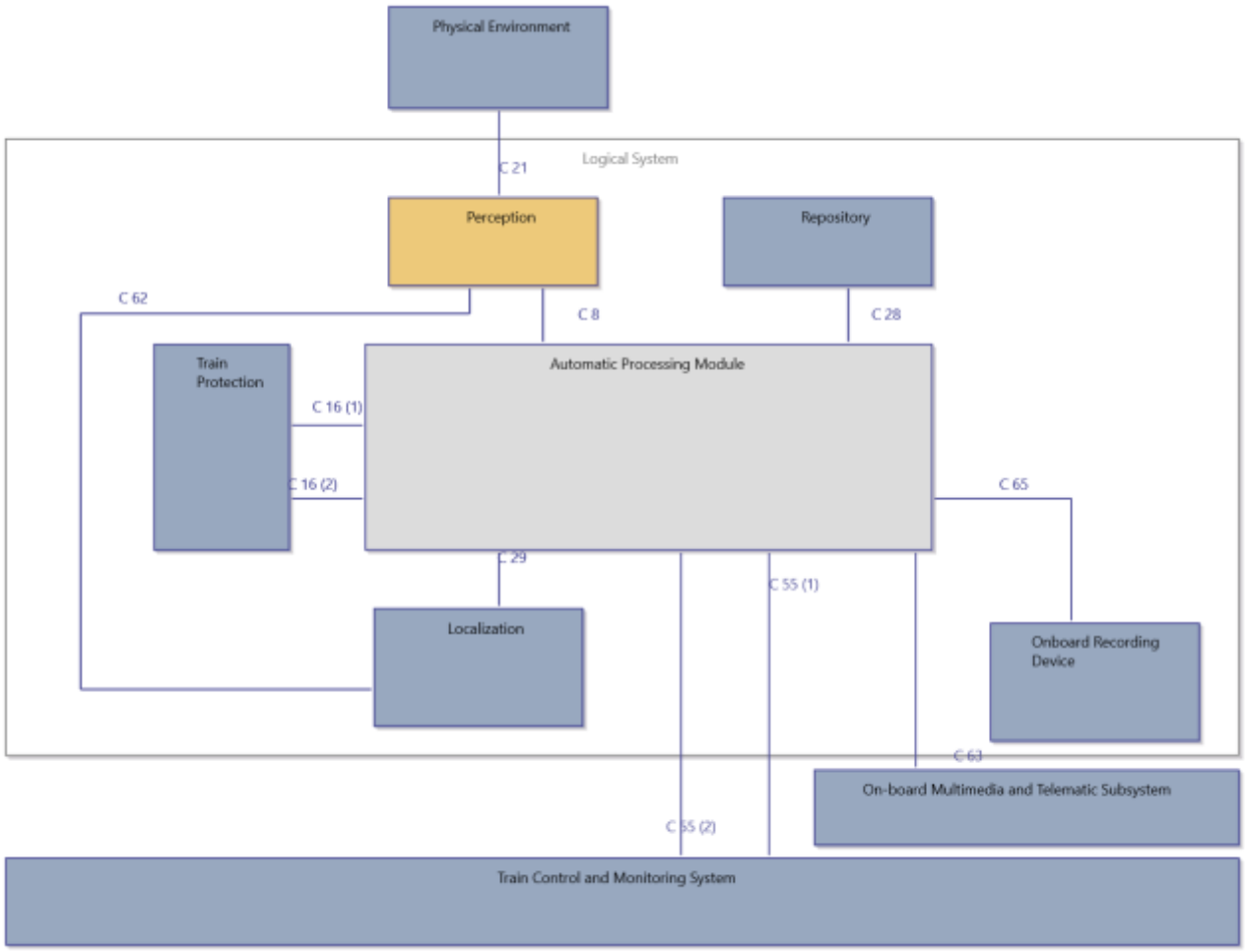

Figure 44: APM and PER test

2.5.2.3 Process Results

For each process, following

- Test Definition, Review

The results of this process step are documented in the project's test case specification, defined by abbreviation [26] in [22].

Chapter 5 provides, per use case, an analysis about the [29] functions needed to implement the use-cases.

Chapter 6 defines decision about the generic test procedure identified as necessary along the project.

Later chapters define, per use case, the associated test case definitions.

- Review

[29] has been provided for review to developers, test designer or WP-lead when the document author and WP lead felt like a consistent body of work was available, and needed it. Comments

Interne

are provided in the requirement management tool until they can be processed in a way satisfying both document and comment authors.

The reviews are archived in the requirement management tool. For instance, the first review was performed after the procedure and the set of test cases addressing UC5.2-0017: Human accident involving injury or death were available.

- Test Configuration

  The results of this process are partly document in [23]: structuring of the test cases in preliminary steps. The detailed test configuration is documented in the project's test report, defined by abbreviation [23] in [25].

- Lab integration, test Implementation,

  Sources and deliveries for Lab integration and test implementation are managed under configuration in GTSD's innovation development environment, as detain in chapter 3 'Compiled deliveries' of [22].

- Test execution and evaluation

  The test implementation is documented in the project's test report, defined by abbreviation [25] in [22].

2.5.3 Feedback for Specification and/or Demonstrators

Early questions and feedback were communicated to the WP10 project leader based on the experiences collected while specifying, implementing or specifying tests for APM, PERception and their test environment. Following table recalls the feedback. Although the project ended with [11] the feedback, made in the early phase of the project, was made on [29].

Document	Reference	Comment	Proposed changed
[27]		Step 4.2.3, Step 4.3.3 refers to "Step 6" but "Step 6" is missing. Our assumption: "Step 5" was meant and not "Step 6".	Replace the reference to "Step 6" by the correct use case step.
[28]	UC5.2-0018: Unusual impact or movement	Not find any packet, variable or value in [30] which corresponds the instability of bogie. Our assumption: C55 (missing in the Excel file) should contain the information corresponding to the instability of bogie.	Please add the related packet(s) in [30]

Interne

Document	Reference	Comment	Proposed changed
[31] [30]	Interface C8, Typo	Discrepancy in packet name between different sources: (plural, singular). LA_GoA34_S2R_1.0.0_20231214.xlsx: PER_APM_OBSTACLE_DETECTIONS Capella Model: PER_APM_OBSTACLE_DETECTION	Correct the name of the packet in LA_GoA34_S2R_1.0.0_20231214.xlsx.
[29]	Interface C8, General	Different incidents are defined but not classified according to their severity. Thus, it is unclear, how the train should react in case of occurrence of more than one incident at the same time. How will the priority be set in case of conflicting incidents? Example case: Child on the track vs. medical urgency (e.g. heart attack) in the compartment.	Define a classification of different incidents according to their severity.
[29] [30]	Interface C8, General	Several values for the variable NC_OBST_CLASS are already defined in packet PER_APM_OBSTACLE_DETECTIONS. However, no value for some other occurrences (e.g. track defects, train defects, compartment incidents) has been defined yet. Are further class types planned for new use cases of R2DATO WP5?	Add further values track defects, train defects, compartment incidents to the variable NC_OBST_CLASS.
[29] [30]	Interface C28, Low-level/technical issues	The packet REP_APM_Segment_Profile_Dynamic given in 10.5.5 C 28 and 10.3.5 C 28 is missing in the LA_GoA34_S2R_1.0.0_20231214.xlsx.	Add the packet REP_APM_Segment_Profile_Dynamic in [30]

Document	Reference	Comment	Proposed changed
[30]	Interface C28, Low-level/technical issues	The variable NC_TRACKCOND_TYPE of the packet REP_APM_SEGMENT_PROFILE_STATIC_PER does not contain any value which corresponds the occurrence of a "flooding".	Add the value "flooding" to the variable NC_TRACKCOND_TYPE.
[30]	Interface C8, Proposal	The variable N_VED_DETECT_ITER of the packet PER_APM_VISUAL_EVENT_DETECTION does not contain some occurrences, such as running children, fighting scene, people under effect of alcohol or drugs. (APM-14.5.1) probably does not help to cover to above mentioned cases.	Add the following attributes: running children, fighting scene, people under effect of alcohol or drugs
[29] [30]	Requirement APM-14.5.1 Low-level/technical issues	[29]'s variable D_OBST_TO_GAUGE in the packets C8.PER_APM_OBSTACLE_DETECTIONS not found in [30] The amendment record sheet of [30] says it was renamed D_OBST_TO_TRACK.	Change D_OBST_TO_GAUGE to D_OBST_TO_TRACK in [29].
[30]	§10.3.7, interface C55 (1) Low-level/technical issues	The packets related to the communication between APM and TCMS (C55) not found. The following have been found: a) LA_GoA34_S2R_1.0.0_20231214.docx specifies in figure 6.1 C55(1) & (2). B) S-139 v1.0.0 specifies: "§5.1.1.1. The scope of this document covers the signals exchanged between ATO-OB and Rolling stock in order to control traction, brakes and doors. Other functionalities (like passenger information system control) are not covered by this document."	Is something missing? Is S-139 also for APM? If so, how are concurrent access solved?

Interne

Document	Reference	Comment	Proposed changed
[29]	Requirements APM-14.8 and APM-13.6	The packets related to following connectors in LA_GoA34_S2R_1.0.0_20231214.xlsx have not been found. - C28 REP-APM - C65 APM-ORD - C63 ORD-APM	If the packet(s) and variables are already defined, be it draft, can a preliminary version be released?

Table 11: Improvement suggestions by GTSD

2.6 HITACHI

Within WP10 the prototypes developed and validated by HITACHI are the ATO GoA3/4 subsystems ADM and REP together with the associated communication interfaces defined in WP6 (aligned with the functional and architectural requirements described in Deliverable D6.5 [11]).

The objective of Task 10.3 was to verify functional completeness, correct interface behaviour, and consistency with the architectural specifications updated in Task 10.1.

The validation was performed using a Software-in-the-Loop (SIL) environments enabling controlled and repeatable testing from TRL4/5, ensuring coverage of both subsystem-level and integrated scenarios.

The results confirmed functional correctness and interface consistency, supporting the preparation of the activities planned in WP39 for integration into the higher-TRL demonstrator.

2.6.1 Methodology

In the context of the ATO up to GoA-4 system defined in the R2DATO project, Hitachi has developed three subsystems: Automatic Driving Module (ADM), Repository (REP) and Automatic Processing Module (APM). The development of the three modules has been performed according to the functions and requirements described within the scope of R2DATO APC Baseline 1 (Release Note FP2-T4_1-I-ATB-047-01) with a particular focus on D6.5 [11].

For the R2DATO project Hitachi has put in place a development and validation process that guarantees the independence between the development team that develops the prototypes and the V&V team that verifies the development artifacts and tests the prototypes.

The test implementation is composed of three independent steps that correspond to the three different high level test environments defined in the context of the R2DATO project:

- First step: APM, REP and ADM are validated separately in a simulated host environment called SimATO Host executed in a single Personal Computer. The SimATO Host Environment is able to simulate the final subsystem HW and can stimulate the interfaces of the subsystems under test according to the simulation steps defined in the test scripts associated to the test specification. The test environment of the first step is Software in the Loop test bench configuration.
- Second step: in this environment APM, REP and ADM are executed on the real targets in a Hardware in the Loop test bench environment. The three subsystems are tested together to verify the integration between the components. The external subsystem interfaces are stimulated with a test tool called SimATO Target Environment-
- Third step: APM, REP and ADM are mounted on the Hitachi Rail Blues train used to validate and integrate the three subsystems in a relevant environment

Test results confirmed the prototype worked and guided ongoing improvements.

This document pertains exclusively to the First Step Test activities conducted for the ADM and REP tasks within the scope of R2DATO WP10.

The First step test activities related to APM are reported in the deliverables of the R2DATO WP9.

Interne

The Second and Third step test activities are part of the Mainline Demonstrator scope of work and are reported in the WP39 deliveries.

The ADM and REP test activities are executed in the environment and tested as defined in the Step 1.

The test execution generates some output files containing both the expected and actual results, as well as the data packets that have been exchanged between interfaces. These outputs are used for checking the correct behaviour of the prototype and for performing a detailed analysis of the data in case of being necessary. The output is used also for generating the test result classified as validated, failed or justified validation (fulfilment justified) with reasoning for compliance.

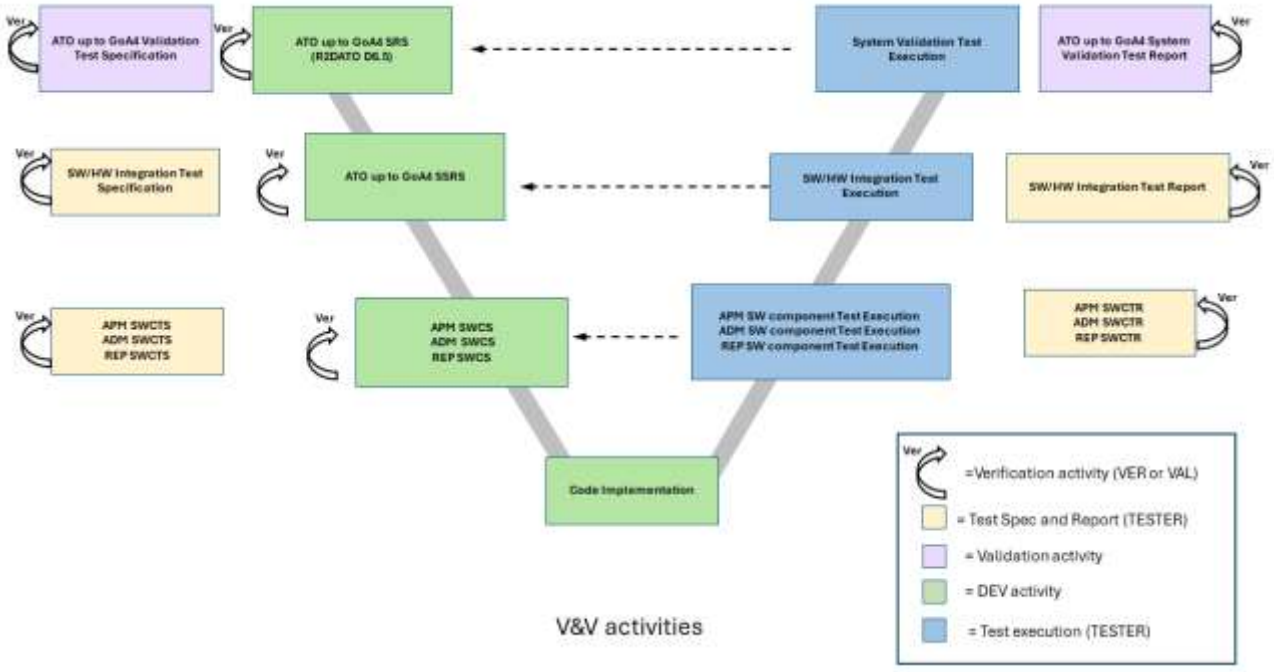

Figure 45: HITACHI process

Diagram shown in Figure 45 describes the development and validation process implemented in the context of the R2DATO project. To ensure the best functional coverage of the requirements, each requirement to be tested is split in unitary test cases. When a requirement involves several conditions that affect a decision, a test case is defined to test the impact of each condition on the decision result (other conditions are unchanged). One or several test cases can be written to test one requirement.

Tests are translated into file (referred as “test scenarios”) in the format accepted by test host environment. A scenario contains a sequence of actions and controls necessary to test one or several test cases. An action is a test input, while a control is a check on the software output. The expected result is described in the scenario (goal).

A test scenario is composed of:

- initial input data;
- actions to generate the inputs on the component under test;
- controls to check the outputs of the component under test;

Interne

- comments to give the test description and the test case traceability;
- results.

When all test scenarios have been implemented, it follows the execution on the test environment. A report and log file are semi-automatically generated for each test scenario. The failed tests are analysed and the anomalies detected are reported in a Work Item.

2.6.2 Test Activities Description

2.6.2.1 Equipment Under Test

The components under test are ADM and REP developed in task 10.1 following the requirements defined in R2DATO APC Baseline 1.

During the testing phase, the equipment utilized consists solely of a testing PC operating the prototype under validation, along with a simulator to generate the required interfaces. Deployment of the prototype on the final hardware platform (TRL 5) is scheduled as part of the WP39 work package.

ADM interfaces involved during the validation process are:

- C22 ADM-REP: data exchange generated by a real REP.
- SS130 ADM-TP: data exchange generated by an emulated TP.
- SS139 ADM-TCMS: data exchange generated by an emulated.

C09 ADM-LOC interface is out of scope; no functional requirements defined for these data.

REP interfaces involved during the validation process are:

- C22 ADM-REP: data exchange generated by a real ADM.
- C28 REP-APM: data exchange generated by a real APM.
- C05 REP-TP: data exchange generated by an emulated TP.
- TRACK TRAIN DATA EXCHANGE (C01, C14, C19, C24, C34) data exchange generated by an emulated Train&Traffic transactor.

C60 REP-LOC interface is out of scope; no functional requirements defined for these data.

In the previous list, two interfaces are marked as “out of scope” because the reference specification baseline does not define functional requirements for the use of their data. As a result, leveraging these interfaces is left to supplier-specific design choices. Within the Hitachi Autonomous Train architecture, Localization is a key component, but it is currently still at an emulation stage as consequence testing these interfaces is considered premature.

Interne

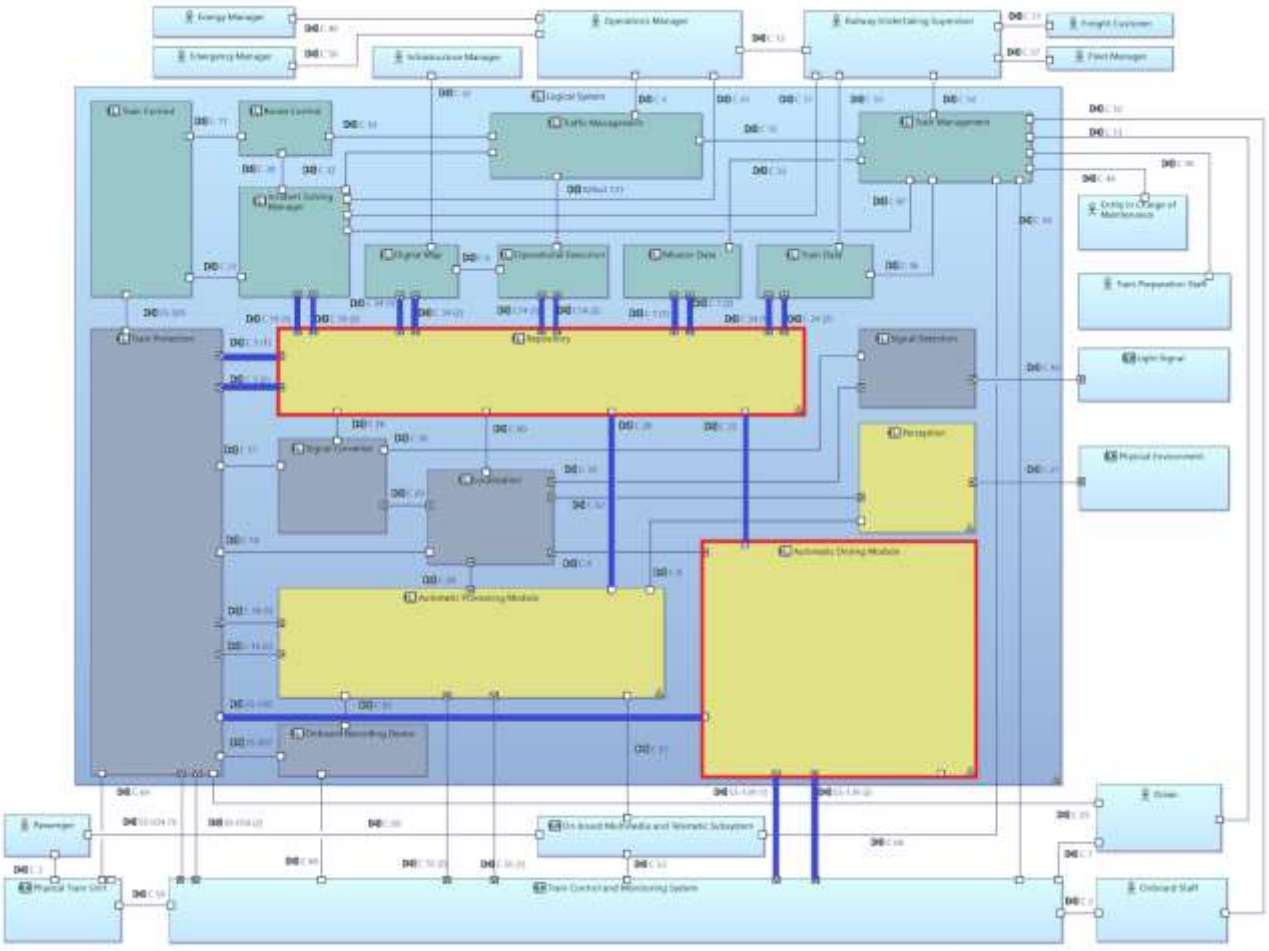

Figure 46: HITACHI Relation between the System and the Equipment under test (with red border)

2.6.2.2 Validation Laboratory Architecture

The scalability of the test environment developed by task 10.1 allows to set up a configuration based on a PC platform (see Figure 47) and permitting prototype testing at functional/requirement level (TRL 4/5).

Interne

Figure 47: HITACHI Target environment GUI

SimATO is the test engine orchestrating the scenario (defined by the validation team) evolution considering the reaction of train dynamic and triggering events (for instance signalling/operational updates). Scenario information are stored in a xml file where track layout is defined as well. A transversal logging function is resident in each prototype making available the collection of proprietary diagnostic data for further test results analysis.

One of the most important configuration capability of SimATO is related to the train model able to emulate train dynamics belonging to different application contexts as Main line, regional and freight. The validation laboratory architectures for ADM and REP validation are illustrated in Figure 48.

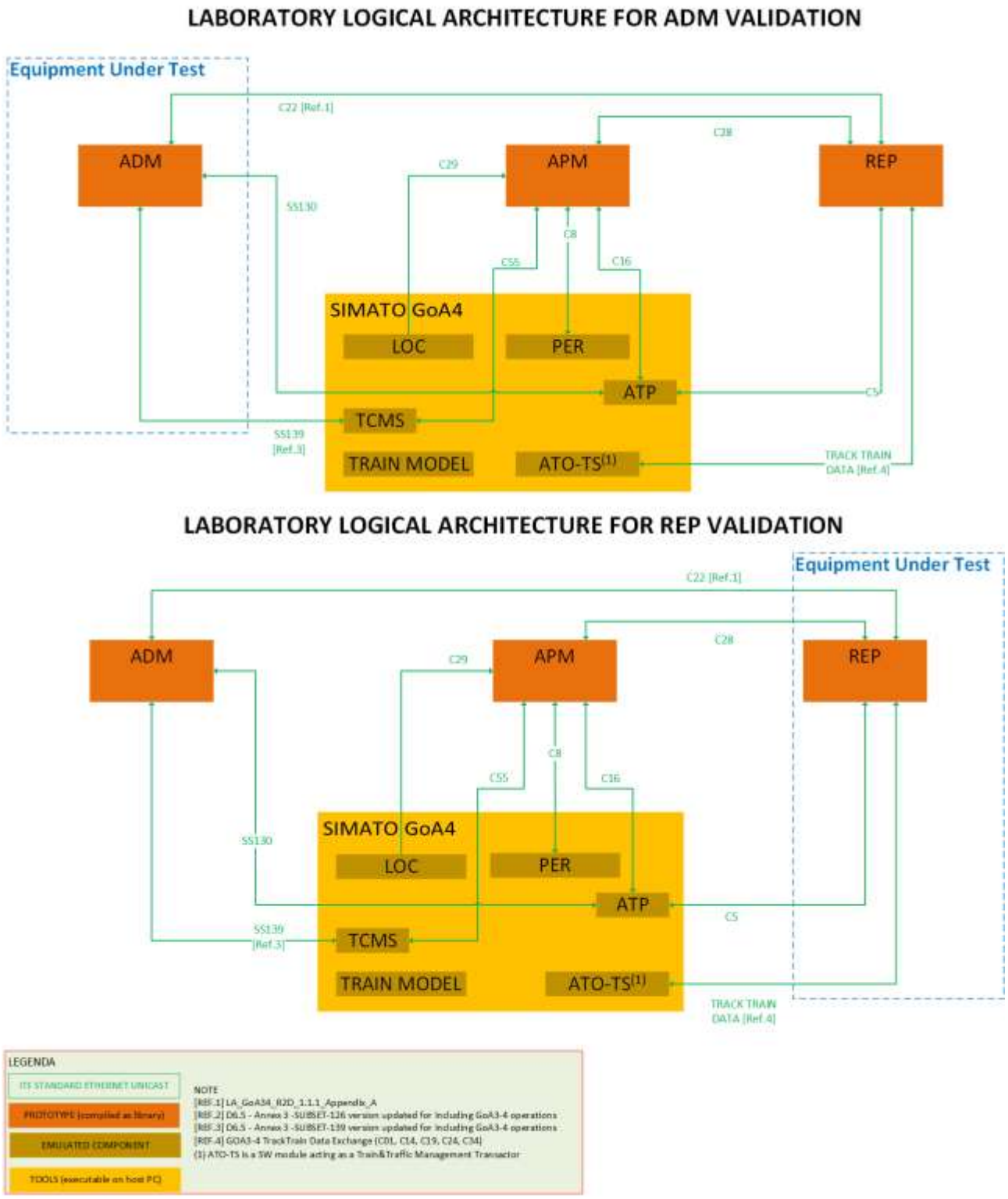

Figure 48: HITACHI Laboratory architectures for ADM and REP validation

The test environment consists of a Windows 11 computer with SimATO installed. The software component being tested is stubbed and compiled as a DLL. It runs within a base software package that simulates the hardware platform and enables interaction between the simulator and the component under test. Specifically, SimATO emulates the base software, which includes the part of

Interne

the component responsible for providing generic APIs to access low-level operating system services. Figure 49 illustrates testing environment.

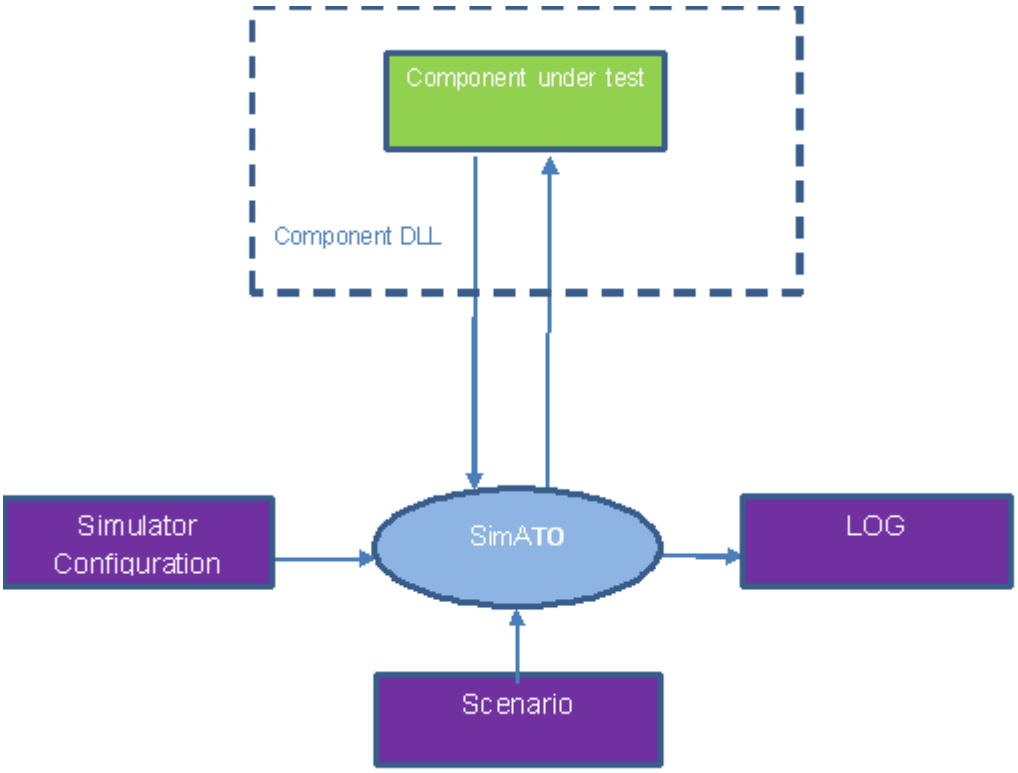

Figure 49: HITACHI Emulation process within SimATO

2.6.2.3 High Level Test Description

The validation team followed a structured process divided into four distinct phases. The first phase focused on verifying the system specifications, which included: (1) the "R2DATO APC Baseline 1" document (FP2-T4_1-I-ATB-047-01), (2) the internal specification titled "ATO up to GoA4 SSRS", and (3) the "Software Component Specification (SWCS)" documents for each individual component—ADM, REP, and APM. The second phase involved the creation of detailed test specifications for each component, including the definition of relevant test scenarios. The third phase was dedicated to executing the tests, while the final phase consisted of compiling the results into comprehensive test reports.

2.6.2.3.1 ADM Validation Results

In accordance with the adopted process, the ADM test specification [17] has been defined. In the Table 12 and Table 13 are listed the tests specified, highlighting the functions under test and the related SRS [16] and Subset 125 v1.0.0 requirements.

Figure 50 provides an indication of the results of the third phase of the process; the detailed results are reported in the test report [18].

TEST ID	FUNCTION	ATO up to GoA4 SSRS
TC_ATO_R2D_315	Take train responsibility when virtual cab is active	REQ_SRS_GoAUpTo4_83

Interne

TEST ID	FUNCTION	ATO up to GoA4 SSRS
TC_ATO_R2D_316	Take train responsibility when virtual cab is active	REQ_SRS_GoAUpTo4_83
TC_ATO_R2D_317	Take train responsibility when virtual cab is active	REQ_SRS_GoAUpTo4_84
TC_ATO_R2D_318	Take train responsibility when virtual cab is active	REQ_SRS_GoAUpTo4_84
TC_ATO_R2D_319	TDO management at wake up start location	REQ_SRS_GoAUpTo4_89
TC_ATO_R2D_320	TDO management at wake up start location	REQ_SRS_GoAUpTo4_89
TC_ATO_R2D_321	ATO Operational Conditions	REQ_SRS_GoAUpTo4_90 REQ_SRS_GoAUpTo4_91
TC_ATO_R2D_322	ATO Operational Conditions	REQ_SRS_GoAUpTo4_95 REQ_SRS_GoAUpTo4_96
TC_ATO_R2D_323	ATO Operational Conditions	REQ_SRS_GoAUpTo4_95
TC_ATO_R2D_324	ATO Operational Conditions	REQ_SRS_GoAUpTo4_95
TC_ATO_R2D_325	ATO Operational Conditions	REQ_SRS_GoAUpTo4_95
TC_ATO_R2D_326	ATO Operational Conditions	REQ_SRS_GoAUpTo4_95 REQ_SRS_GoAUpTo4_96
TC_ATO_R2D_327	FSM Evolution	REQ_SRS_GoAUpTo4_97
TC_ATO_R2D_328	FSM Evolution	REQ_SRS_GoAUpTo4_97
TC_ATO_R2D_329	FSM Evolution	REQ_SRS_GoAUpTo4_100
TC_ATO_R2D_330	FSM Evolution	REQ_SRS_GoAUpTo4_100
TC_ATO_R2D_331	FSM Evolution	REQ_SRS_GoAUpTo4_100
TC_ATO_R2D_332	TDO at EoJ	REQ_SRS_GoAUpTo4_109
TC_ATO_R2D_333	TDO at EoJ	REQ_SRS_GoAUpTo4_109
TC_ATO_R2D_334	JP Completed	REQ_SRS_GoAUpTo4_110 REQ_SRS_GoAUpTo4_136 a) REQ_SRS_GoAUpTo4_184
TC_ATO_R2D_335	JP Completed	REQ_SRS_GoAUpTo4_110 REQ_SRS_GoAUpTo4_136 a) REQ_SRS_GoAUpTo4_184
TC_ATO_R2D_336	Interface	REQ_SRS_GoAUpTo4_126
TC_ATO_R2D_337	Interface	REQ_SRS_GoAUpTo4_143
TC_ATO_R2D_338	Interface	REQ_SRS_GoAUpTo4_143
TC_ATO_R2D_339	Interface	REQ_SRS_GoAUpTo4_143
TC_ATO_R2D_340	Interface	REQ_SRS_GoAUpTo4_143
TC_ATO_R2D_341	Interface	REQ_SRS_GoAUpTo4_143
TC_ATO_R2D_342	Interface	REQ_SRS_GoAUpTo4_143
TC_ATO_R2D_343	Interface	REQ_SRS_GoAUpTo4_136 b)

TEST ID	FUNCTION	ATO up to GoA4 SSRS
TC_ATO_R2D_344	Interface	REQ_SRS_GoAUpTo4_136 d)
TC_ATO_R2D_345	Interface	REQ_SRS_GoAUpTo4_136 d)
TC_ATO_R2D_346	Interface	REQ_SRS_GoAUpTo4_136 e)
TC_ATO_R2D_347	Interface	REQ_SRS_GoAUpTo4_136 e)
TC_ATO_R2D_348	Take train responsibility when virtual cab is active	REQ_SRS_GoAUpTo4_87
TC_ATO_R2D_349	Take train responsibility when virtual cab is active	REQ_SRS_GoAUpTo4_87
TC_ATO_R2D_350	Communication session establishment	REQ_SRS_GoAUpTo4_121
TC_ATO_R2D_351	Communication session establishment	REQ_SRS_GoAUpTo4_121
TC_ATO_R2D_352	Interface	REQ_SRS_GoAUpTo4_125

Table 12: HITACHI List of test defined for ADM with reference to requirements of "ATO up to GoA4 SSRS"

TEST ID	FUNCTION	SS125
TC_ATO_R2D_020	ATO System Version Management	7.9
TC_ATO_R2D_003	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_008	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_032	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_039	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_071	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_072	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_073	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_074	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_075	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_076	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_077	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_078	Compute the ATO Operational Speed Profile	7.1.2

TC_ATO_R2D_079	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_082	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_084	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_085	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_086	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_088	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_092	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_093	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_115	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_140	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_143	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_148	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_211	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_220	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_233	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_238	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_308	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_310	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_312	Compute the ATO Operational Speed Profile	7.1.2
TC_ATO_R2D_004	Compute the dwell time	7.2.3
TC_ATO_R2D_003	Compute the dwell time	7.2.3
TC_ATO_R2D_012	Detect data inconsistency	7.8
TC_ATO_R2D_013	Detect data inconsistency	7.8
TC_ATO_R2D_014	Detect data inconsistency	7.8
TC_ATO_R2D_015	Detect data inconsistency	7.8
TC_ATO_R2D_016	Detect data inconsistency	7.8
TC_ATO_R2D_017	Detect data inconsistency	7.8
TC_ATO_R2D_018	Detect data inconsistency	7.8
TC_ATO_R2D_019	Detect data inconsistency	7.8
TC_ATO_R2D_004	Hold train at a Stopping Point	7.2.3.3
TC_ATO_R2D_003	Hold train at a Stopping Point	7.2.3.3

Interne

TC_ATO_R2D_005	Manage changes in Stopping Points in the JP	7.3
TC_ATO_R2D_006	Manage changes in Stopping Points in the JP	7.3
TC_ATO_R2D_007	Manage changes in Stopping Points in the JP	7.3
TC_ATO_R2D_107	Manage changes in Stopping Points in the JP	7.3
TC_ATO_R2D_172	Manage changes in Stopping Points in the JP	7.3
TC_ATO_R2D_144	Manage low adhesion	7.5
TC_ATO_R2D_071	Manage low adhesion	7.5
TC_ATO_R2D_057	Manage low adhesion	7.5
TC_ATO_R2D_002	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_003	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_005	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_007	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_012	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_013	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_014	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_015	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_016	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_017	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_018	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_019	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_028	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_057	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_098	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_099	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_127	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_133	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_144	Provide STRs to ATO-TS	7.7
TC_ATO_R2D_009	Send train doors opening/closing requests	7.2.2
TC_ATO_R2D_010	Send train doors opening/closing requests	7.2.2
TC_ATO_R2D_268	Send train doors opening/closing requests	7.2.2
TC_ATO_R2D_134	SP version management	10.1.7.25
TC_ATO_R2D_017	SP version management	10.1.7.25
TC_ATO_R2D_011	Time Management	7.6

Table 13: HITACHI List of test defined for ADM with reference to requirements of SS125

Interne

Figure 50: HITACHI ADM test results

2.6.2.3.2 REP Validation Results

In accordance with the adopted process, the REP test specification [19] has been defined. In the Table 14 are listed the tests specified, highlighting the functions under test and the related SRS requirements [16].

Figure 51 provides an indication of the results of the third phase of the process; the detailed results are reported in the test report [20].

TEST ID	FUNCTION	ATO up to GoA4 SSRS
TC_R2D_REP_001	Acquire MP	REQ_SRS_GoAUpTo4_155
TC_R2D_REP_001	C05 interface/Overview	REQ_SRS_GoAUpTo4_325 REQ_SRS_GoAUpTo4_333
TC_R2D_REP_001	C22 Interface/Manage session	REQ_SRS_GoAUpTo4_123 REQ_SRS_GoAUpTo4_265 REQ_SRS_GoAUpTo4_266 REQ_SRS_GoAUpTo4_267 REQ_SRS_GoAUpTo4_268 REQ_SRS_GoAUpTo4_122
TC_R2D_REP_001	C22 Interface/Overview	REQ_SRS_GoAUpTo4_244
TC_R2D_REP_001	C28 interface/Manage communication	REQ_SRS_GoAUpTo4_310 REQ_SRS_GoAUpTo4_316 REQ_SRS_GoAUpTo4_317 REQ_SRS_GoAUpTo4_318
TC_R2D_REP_001	C28 interface/Overview	REQ_SRS_GoAUpTo4_309
TC_R2D_REP_001	Determine REP state	REQ_SRS_GoAUpTo4_404 REQ_SRS_GoAUpTo4_405 REQ_SRS_GoAUpTo4_407 REQ_SRS_GoAUpTo4_414 REQ_SRS_GoAUpTo4_416
TC_R2D_REP_001	Establishing comm. session	REQ_SRS_GoAUpTo4_208 REQ_SRS_GoAUpTo4_209
TC_R2D_REP_001	Establishing comm. session/Handshake & Versioning	REQ_SRS_GoAUpTo4_215 REQ_SRS_GoAUpTo4_216 REQ_SRS_GoAUpTo4_246 REQ_SRS_GoAUpTo4_247

Interne

TEST ID	FUNCTION	ATO up to GoA4 SSRS
TC_R2D_REP_001	Manage Reporting (RU)\APM interaction	REQ_SRS_GoAUpTo4_359 REQ_SRS_GoAUpTo4_370
TC_R2D_REP_001	Transactor Contact Information	REQ_SRS_GoAUpTo4_198
TC_R2D_REP_002	Manage Reporting (RU)\APM interaction	REQ_SRS_GoAUpTo4_363 (Covered by 1k) REQ_SRS_GoAUpTo4_364 (Covered by 1 a),b),c),d),e),f) g) REQ_SRS_GoAUpTo4_365 (Covered by 1 h) REQ_SRS_GoAUpTo4_366 (Covered by 1 i) REQ_SRS_GoAUpTo4_367 (Covered by 1 j) REQ_SRS_GoAUpTo4_368 (Covered by l),m) n) REQ_SRS_GoAUpTo4_364 (Covered by 1 a),b),c),d),e),f) g) REQ_SRS_GoAUpTo4_365 (Covered by 1 h) REQ_SRS_GoAUpTo4_366 (Covered by 1 i) REQ_SRS_GoAUpTo4_367 (Covered by 1 j) REQ_SRS_GoAUpTo4_368 (Covered by l),m) n)
TC_R2D_REP_004	Manage Reporting (IM)\ADM interaction	REQ_SRS_GoAUpTo4_53
TC_R2D_REP_004	Establishing comm. session /Interface	REQ_SRS_GoAUpTo4_212
TC_R2D_REP_004	Establishing comm. session	REQ_SRS_GoAUpTo4_213
TC_R2D_REP_004	Maintaining a communication session	REQ_SRS_GoAUpTo4_254 REQ_SRS_GoAUpTo4_257
TC_R2D_REP_004	Manage Reporting (RU)\APM interaction	REQ_SRS_GoAUpTo4_369
TC_R2D_REP_005	Loss of communication	REQ_SRS_GoAUpTo4_236
TC_R2D_REP_005	Establishing comm. session/Handshake & Versioning	REQ_SRS_GoAUpTo4_228
TC_R2D_REP_006	Establishing comm. session/Handshake & Versioning	REQ_SRS_GoAUpTo4_236 REQ_SRS_GoAUpTo4_228
TC_R2D_REP_007	Establishing comm. session/Handshake & Versioning	REQ_SRS_GoAUpTo4_236
TC_R2D_REP_010	Manage Reporting (RU)\APM interaction	REQ_SRS_GoAUpTo4_361

TEST ID	FUNCTION	ATO up to GoA4 SSRS
TC_R2D_REP_011	Manage Reporting (RU)\APM interaction	REQ_SRS_GoAUpTo4_362
TC_R2D_REP_012	C28 interface/Overview	REQ_SRS_GoAUpTo4_311
TC_R2D_REP_013	C28 interface/Overview	REQ_SRS_GoAUpTo4_311
TC_R2D_REP_014	Manage Reporting (ISM)\APM interaction	REQ_SRS_GoAUpTo4_379
TC_R2D_REP_015	Acquire train parameters	REQ_SRS_GoAUpTo4_342 REQ_SRS_GoAUpTo4_350
TC_R2D_REP_016	Map REP with Route	REQ_SRS_GoAUpTo4_438
TC_R2D_REP_017	Acquire instructions	REQ_SRS_GoAUpTo4_400

Table 14: HITACHI List of test defined for REP with reference to requirements of "ATO up to GoA4 SSRS"

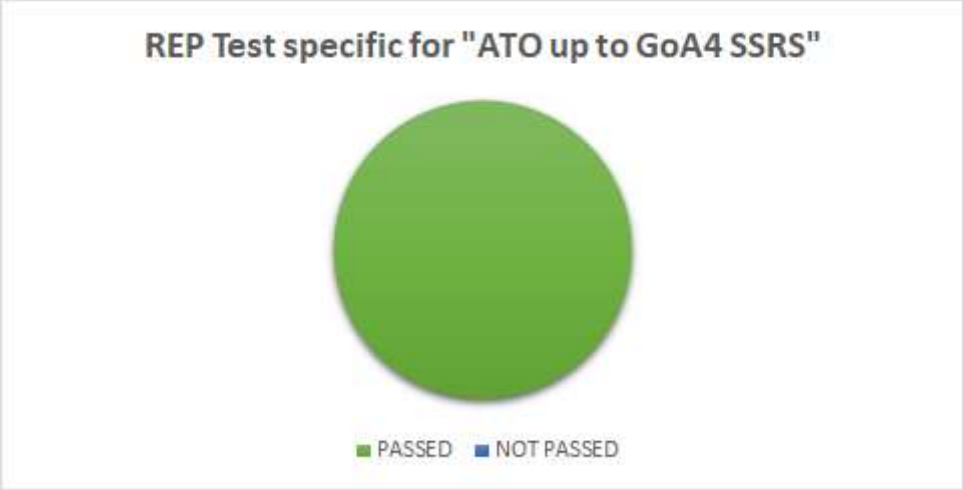

Figure 51: HITACHI REP test results

2.6.3 Feedback for Specification and/or Demonstrators

After test specification execution, it is possible to confirm that the functions and requirements specified in the T6.5 deliverable GoA34_R2D_1.1.1 document for both of our prototypes, Repository (REP) and Automatic Driving Module (ADM), are validated as far as needed for our analysis and component requirements.

The specification provided by R2DATO have permitted to realize prototypes for train automatic operation up to GoA4 in nominal condition and also considering some non-regular situations. Mechanism to retrieve information from track side have been found in some parts incomplete and a proposal of improvement is shown in Table 15. Some requirements needs a new formulation as proposed in Table 16. Interfaces needs also improvements as outlined in Table 17.

ID	Reference	Proposal	Rational
100	15.2.3	Add here the list of packets and their definition dedicated for session management	Session management between REP and transactor are independent respect to the type of service (role) provided by the transactor
101	15.3.6.1.7	T_Timeout_ATOTS_Response shall be sent to the value received by Traffic Transactor	
102	15.3.6.1.8	T_Reporting_Time shall be sent to the value received by Traffic Transactor	
103	15.3.7.3.2	Rewrite the clause to be aligned with 8.6.1.1.7	
104	15.3.7.4		
105	15.3.7.2.10	Replace “REP shall stop trying” with “APM shall stop trying”	Is APM in charge to open connection
106	15.3.7.1.9	Remove or Change to explain that over interface C28 there is no need of an handshake process	The functions of APM are not dependent by a session with REP (some approach of ss130)

Table 15: HITACHI List of improvements suggested to stabilize REP communication principles

ID	Reference	Proposal	Rational
1	10.5.9 & 10.5.10	Remove session management packet	C14 interface as the others shall contain only application packets
2	10.5.10	Remove EI “Segment Profile dynamic data”	No added value in removing dynamic data from JP and putting in a new dedicated packet.
3	10.5.16	Allocate FE “task delay warning”, “task rejected” in the EI “REP_MD_Status_Report”	These FE reflects the status of the mission
4	10.5.17	Add EI “TD_VALID_REQ”	This EI is needed to cover the case of trains without a second source of Train Data (see ppt meeting #52 07/11/24)
5	10.5.13	Double EI ISM_REP_Acknowledgement one for Track anomaly and one for Train anomaly	By reusing the basic structure of acknowledgment is more

ID	Reference	Proposal	Rational
			convenient to have a dedicated ack packet.
6	10.5.15	Remove EI "REP_ISM_Obstacle_Detected" and move FE "Obstacle detected" in EI "REP_ISM_Track_Incident_Report"	Just to improve organization of information exchanged within the interface
7	REP-2.4	Remove the first part of the sentence: "In GoA3/4 operation and when REP is in OP State,"	MP is the trigger for starting any kind of ATO up to GoA3/4 activity, can not be constrained to an "advanced" state of REP.
8		Add requirement in 12.13.1 (MD-1.17) The mission profile of a train planned to be operated in GoA2, shall contain the list of TJPs providing Train Running Number (NID_REFERENCE_TRID) and Traffic Management Transactor Contact Information (NID_C and NID_ TRAFFICTRANSACTOR).	Need of adding a requirement in the composition of MP in case of TJP planned to be operated in GoA2. The connection is allocated to REP and the rules I chapter 15 oblige the connection to traffic transactor via the information received in the MP.
9	Rep-2.4	Remove condition on REP state	MP shall even be sent after wake up when REP is in NA
10	TP-4.1	Replace APM with REP	APM is already in GoA3/4
11	8.6.4.8	Add a requirement telling that REP provides "Map MA with route" information when in OP	
12	8.6.4.9.2	Not defined how the driver shall be informed about REP failure	

Table 16: HITACHI List of improvements suggested to stabilize ADM and REP requirements

ID	Reference	Proposal	Rational
1000	Appendix C28 REP_APM_MISSION_ PROFILE T_TASK_START_TIME and T_TASK_END_TIME	Replace description with "Start time of Task in UTC time (The 15 MSBs represent number of days and the 17 LSBs represent number of seconds from the 1st January 2010.)"	Current description doesn't give enough information

ID	Reference	Proposal	Rational
1001	Appendix C28 REP_APM_MISSION_ PROFILE	Adding a field to provide NID_C of the transactor to be contacted for retrieving JP information	NID_TRANSACTOR is not suffcient
1002	Appendix C28 REP_APM_SEGMENT_ PROFILE_STATIC_PER	Replace description with “Number of zones within the segment with a common pair of values of "Track zone" or" Near Track zone" width.” Replace Values with: “1 - 31 0 = spare [32-255] = spare”	At least one pair of "Track zone" or" Near Track zone" width.” Shall be defined for the segment
1003	Appendix C28 REP_APM_MISSION_ PROFILE	Move field NID_TD_TRAINUNIT outside iteration cycle	MP is made at train unit level as soon as a split or coupling is performed the MP shall be overwritten. See req. on MP in MD entity
1004	Appendix C28 APM_REP_STATUS_RE PORT	Add the following type of train anomalies detected by TCMS: 16 = Speed reduction 17 = Operational limitation: immediate stop managed by TCMS 18 = Operational limitation: immediate speed reduction managed by TCMS 19 = Operational limitation: stop train as soon possible 20 = Operational limitation: Remove train from operation	Reason REP need these events in order to forward towards ISM or MD
1005	Appendix sheet “Packet_list”	Replace “process data” with “message data” for C28 packets	These packets are sent on an event-base trigger no need for cyclic tx. In addition packet Tx shall be guaranteed
1006	Appendix sheet “Packet_list”	Replace “100” with “REP” for C28 packets	These packets are sent on an event-base trigger no need for cyclic tx. In addition packet Tx shall be guaranteed
1007	Appendix sheet “Packet_list”	Replace “10” with “NA” for C28 packets	These packets are sent on an event-base trigger no need for cyclic tx. In addition packet Tx shall be guaranteed

Interne

ID	Reference	Proposal	Rational
1008	Appendix packet APM_REP_MISSION_STATUS	Add fields as described in [10]	
1009	Appendix packet APM_REP_MISSION_PROFILE	Field 26: replace variable name, description…vaues with "NID_TRAFFICTRANSACTOR" as defined in [10]	
1010	Appendix packet C55 and C28	Add iteration for field M_TCMS_REPORTED_ANOMALY in pkt TCMS_APM_MSG_INCIDENT	Incidents can happen simultaneously
1011	Appendix packet APM_REP_MISSION_PROFILE	Add field NID_C prior to NID_TRAFFICTRANSACTOR	
1012	Appendix A: Packet list	PACKET LIST: updated note 3 to explain which packet of SS126 are applicable for C28 interface PACKET LIST: add in cell A36 note 3 PACKET LIST: replaced note 3 with note 1 in cells: E36, F36, G36, I36, J36	
1013	Appendix A	REP_APM_INFORMATION: field M_DRIVING_ANOMALY added value 255 = no driving anomaly present	
1014	Appendix A C05	Added packet ETCS_REP_LASTMA	channel monitoring. If MA is changed because of a rerouting and this MA is not received by REP, PER could be induced in using a wrong ego track REP shall go in FA if REP is not able to receive MA from Train Protection
1015	Appendix A REP_APM_REP_STATUS	Add value for state NL	
1016	Appendix A APM_REP_MISSION_STATUS	Add a variable to indicate the need of a MP update	

Table 17: HITACHI List of improvements suggested to stabilize ADM and REP interfaces

2.7 SMO

2.7.1 Methodology

Based on the refined requirements in the System Requirements Specification (SRS), test cases are specified in the System Requirements Test Specification (SRTS), where test cases are traced to the requirements from the SRS. For each specified test a test case is generated, which runs on the test environment and stimulates the system under test. For each system a separate test specification is created, since the emulated entities stimulating the system under test differ.

The main objective of the integration and testing is to demonstrate the correct functioning of the overall system in interaction with all external components and actors, in accordance with the respective requirement and architecture specifications and interface specifications.

To achieve the desired test goal, the individual components are tested at software and system level at their respective internal and external interfaces against the corresponding specifications using suitable test tools in a test environment that can reproduce all transactions on the respective interfaces.

The results of the tests on subsystem and system levels are recorded in a common report document. Results of manual tests are directly entered into the report's respective chapters. Results of automated test runs are generated by the test environment.

The testing work is distributed in the SMO sites Berlin and Budapest. A laboratory test environment for the entire AVVO system is established at both sites.

2.7.2 Test Activities Description

In the present section a description of test activities are provided, the associated internal test report e and test plan ([21]) is available for audit purposes only.

For each system of the prototype, a dedicated environment allows testing the different interfaces (see Figure 52).

Interne

Figure 52: SMO 19-inch rack for testing the ATO-OB (left) and the AVVO-OB (right)

The system under test runs on the target hardware (in a 19-inch rack). The interface partners are simulated in the test environment and allow a continuous, real-time stimulation of the system under test. Based on this set-up, different scenarios for testing can be simulated, to cover a wide range of test cases for the system.

The equipment can be operated at normal indoor room temperature. There is no need for specific climatization. The operating staff doesn't need specific wear (like isolation gown, strip, shoes) for electrical isolation.

2.7.2.1 Test Results

The functions described in Task 10.1 have been tested by SMO as follows:

2.7.2.1.1 Determine Maximum Authorised Speed

The function has been implemented and successfully tested.

2.7.2.1.2 Respect JP Timing Points and Optimize the Consumption

The function has been implemented and successfully tested.

2.7.2.1.3 Stop Exactly at the Intended Location

The function has been implemented and successfully tested.

2.7.2.1.4 Compute Operational Speed and Acceleration

Interne

The function has been implemented and successfully tested.

2.7.2.1.5 Regulate Traction and Braking Effort
The function has been partially tested. Some test results are outstanding at the time of writing this report.

2.7.2.1.6 Request Holding Brake
The function has been implemented and successfully tested.

2.7.2.1.7 Manage Low Adhesion
The function has been implemented and successfully tested.

2.7.2.1.8 Acquire JP
The function has been implemented and successfully tested.

2.7.2.1.9 Acquire MP
The function has been implemented and successfully tested.

2.7.2.1.10 Acquire Segment Profile Information
The function has been implemented and successfully tested.

2.7.3 Feedback for Specification and/or Demonstrators

Since the prototypes of WP9 and WP10 overlap on the system level and are only differentiated on functional and use case level, the feedback to specifications from both work packages is similar.

From SMO perspective, the most relevant feedback for the specifications is the lack of a common and more importantly systematic approach for defining the requirements for the systems. The differences in requirements between Task 6.5 and Task 6.1, as well as the lack of definition split between the functions of “Automating Functions” and “ATO GoA3-4 functions” lead to a lack of coherence and clarity giving much room for interpretation. The systematic approach followed by the Task 6.2, starting at operational level and breaking the specifications down to the systems solves these unclarities and brings structure to the approach. Unfortunately, the results of Task 6.2 were too late to be considered in the prototype development of Task 9.2 but must be considered as a starting point for R2DATO Wave 2.

Moreover, reaction to obstacles detected is regulated directly between AVVO-OB, ATO-OB and TCMS, and does not result in modifications to EoA. This design decision was taken mainly to prevent modification of the EVC and any undesired impact to the already standardised ETCS function in general and to consider the specific safety goals of protection against collisions with detected obstacles.

Furthermore, for the function acquire train parameters, the verification of the validity of train data is done at trackside and not onboard. Rationale: the train data cannot be validated onboard alone if the onboard components do not have the “ground truth” for the train data to validate the input. Nonetheless, a live check against current train data determined by the TCMS is conducted by the AVVO-OB and in case of deviations, the more restrictive values are propagated to the EVC.

In addition, in the SMO prototype, the Mission Profile consists of a list of parameters necessary for conducting the mission. Since the mission tasks describe processes that have a rigid structure, the

Interne

exchange of these tasks over the airgap interface does not add any value for the performance of the system and is rather counterproductive. These processes have degrees of freedom that are parametrised within the Mission Profile.

Finally, the prototypical implementation does not realise the monolithic repository (REP) as it is an inhomogeneous group of functionalities. Rather, it is recommended to consider the logical component “REP” as a set of functionalities, which can be taken over by relevant components in the system.

3 CONCLUSIONS

The activities carried out within Deliverable D10.3 have successfully addressed the primary objective of validating ATO prototypes up to GoA3/4 against the functional and architectural specifications defined in WP6 (D6.5) which were developed and upgraded in Task 10.1. The work carried out in parallel by the supplier partners for each prototype has established a structured and coherent testing framework, enabling the verification of interoperability, functional completeness, and overall readiness for integration into future demonstrators. This has been verified in seven different test environments of different suppliers and railway undertakings, ranging from TRL4 simulated environments to TRL6 demonstrators in operational environments.

The main technical contributions can be summarized as follows:

- Definition of a multi-layer validation approaches combining Software-in-the-Loop (SIL), Hardware-in-the-Loop (HIL), and demonstrator-level testing, ensuring coverage from TRL4 to TRL6.
- Successful validation of core functionalities of ATO GoA3/4 such as ADM, APM, REP and PER, and the communication interfaces between these systems. This ensures that ATO solutions meet the functional requirements.
- Consolidation of feedback loops for specification refinement, resulting in concrete proposals for improvement of interface definitions, packet structures, and operational sequences.

The validation outcomes confirm that the prototypes meet the functional requirements under controlled conditions and are suitable for progression toward operational demonstrators. The work strengthens the alignment between ATO and ETCS subsystems, paving the way for cross-vendor interoperability and standard evolution. It also proves the suitability of applying test bench environments for testing the most important ATO (sub-)systems, reducing the required test effort on expensive and scarce physical railway assets.

Several challenges were also encountered:

- Interface ambiguities and inconsistencies forced partners to make their own assumptions, which may differ across implementations, highlighted the need for a formal conformance testing suite and a centralized data catalog. Safety-critical behaviors such as re-engagement after unscheduled stops and differentiation between signal-based and obstacle-based EoAs, are not sufficiently detailed in the current specifications.
- Integration of perception systems still demands more robust incident severity classification and arbitration logic, particularly for concurrent detection events.
- Limitations in TRL6 demonstrator environments prevented full assessment of specific functions, requiring justified validation approaches.

- Developments and testing activities were carried out based on the initial specification baseline produced by the R2DATO, resulting in natural differences in timing between prototype evolution, specification refinement, and demonstrator preparation. This limited the ability to fully synchronize test expectations and system maturity levels.

It is recommended to address these topics within the second wave of R2DATO, both in the updating of the ATO GoA3/4 specification, as well as in the definition of future test environments for prototypes and field demonstrators.

In conclusion, D10.3 provides a solid technical foundation for the industrialization, standardization, and market deployment of ATO GoA3/4 systems. It validates key system functionalities, identifies specification gaps, and proposes actionable improvements. The work supports the strategic objective of interoperability and safety, supporting the path toward large-scale deployment of automated train operations across European rail networks.

4 CROSS-PARTNER FINDINGS AND PROPOSED AMENDMENTS TO THE ATO GOA3/4 SPECIFICATIONS

This chapter consolidates the common findings identified independently by multiple partners during the validation activities of Task 10.3. Although each contributor provided detailed feedback specific to their prototype, test environment, or interpretation of the specifications, several cross-cutting issues emerged consistently across the consortium. These joint findings highlight structural gaps in the current ATO GoA3/4 specifications and provide guidance for future refinement activities.

1. **Need for clearer, complete and stable specifications**

   Multiple partners (ATSA, CAF, FS, GTSD, HITACHI, SMO) reported that several requirements in the SRS and interface definitions need to be improved to solve ambiguities, incompleteness or inconsistency. Examples include unclear packet definitions, duplicated fields, missing variables, and insufficiently defined operational sequences.

2. **Interface update**

   A recurring theme across partners concerns the need to stabilize interfaces (C22, C28, C55, SS130, SS139, etc.).

   Issues include:

   - missing packet definitions or mismatches [12] respect to SRS [11],
   - inconsistent naming conventions,
   - unclear session-management rules.

   These inconsistencies complicate interoperability and require a centralized, harmonized interface catalogue.

3. **Need for improved definition of operational sequences and degraded modes**

   ATSA, AZD, FS, HITACHI and SMO provide recommendations on the type of reaction to be applied in case of degraded scenarios/incidents, including:

   - engagement logic after unscheduled stops,
   - behaviour in non-FS ETCS modes,
   - handling of emergency stops and obstacle-related EoAs,
   - fallback procedures when data is missing or inconsistent.

4. **Improved definition of on Mission Profile (MP) and Journey Profile (JP) management**

   CAF, HITACHI, FS and SMO report:

   - MP update rules,
   - handling of unavailable or invalid tasks,
   - synchronization of MP/JP across multiple units in a consist,
   - definition of special locations (e.g., rescue points).

5. **Need for improved alignment between ATO and ETCS behaviours**

Interne

Several partners (AZD, FS, HITACHI, SMO) propose improvements in the interactions between ATO GoA3/4 and ETCS, particularly regarding:

- behaviour respect to obstacle-based EoAs,
- speed and distance monitoring information,
- consistency of odometry and positioning references.

A coordinated evolution of SS125, SS130 and the GoA3/4 SRS is needed to ensure coherent behaviour.

6. **Perception system integration requires clearer rules**

AZD, GTSD, SMO and CAF identified the need for:

- arbitration rules for concurrent detections,
- alternative mapping between perception outputs and APM reactions.

This is essential to avoid inconsistent or unsafe reactions in complex scenarios.

Interne

5 REFERENCES

[1] ATO GoA4 Baseline 1.6.0 System Requirement Test Report – AoE_GEN_VV_DESG_1155 V1.2

[2] test definitions, scope, and test report for the ADM subsystem - ADM_alcance_test_plan_y_report_v_1.1.1.xlsx

[3] test definitions, scope, and test report for the APM subsystem- APM_alcance_test_plan_y_report_v_1.1.1.xlsx

[4] overall testing scope and general definitions - Alcance_y_descripción_general_testing_v_1.1.1.docx

[5] D10.3 – FS ATO GoA34 System Test Specification and Report - Annex 1 - Subsystems Requirements Specification – Rev A - RFI RESSS ST 06 04 001 0

[6] D10.3 – FS ATO GoA34 System Test Specification and Report - Annex 2 - Design Description – Rev A - RFI RESSS RT 06 04 000 0

[7] D10.3 – FS ATO GoA34 System Test Specification and Report - Annex 3 - Test Specification - Rev A - RFI RESSS SS 06 04 000 0

[8] D10.3 – FS ATO GoA34 System Test Specification and Report - Annex 4 - Test Report - Rev A - RFI RESSS RE 06 04 000 0

[9] EN 50761 - Railway Applications - Requirements for software development

[10] Deliverable D6.5 – ATO GoA3-4 Specifications and modelling – ERJU FP2 R2DATO

[11] D6.5 – Annex 1 – SRS ATO up to GoA3/4 - FP2-T6_5-T-STS-046-04

[12] D6.5 – Appendix A – GOA3/4 PACKETS - FP2-T6_5-T-STS-059-02

[13] D6.1 – Report on sets of non-functional and functional requirements (SRS) for automating functions.

[14] S. Iwnicki, Handbook of Railway Vehicle Dynamics. Boca Raton, FL, USA: CRC Press, 2006

[15] Deliverable D5.3 GoA3/4 Reference Tests Bench Public Report – Shift2Rail X2Rail-4

[16] ERTMS/ATO UP TO GOA3/4 System Requirements Specification – B60X.0100931.S04.00EN

[17] Train Automation Technologies ADM Validation Test Specification – B60X.0100931.S06.00EN

[18] Train Automation Technologies ADM Validation Test Report – B60X.0100931.R07.00EN

[19] Train Automation Technologies REP Validation Test Specification – B60X.0100931.S06.01EN

[20] Train Automation Technologies REP Validation Test Report – B60X.0100931.R07.01EN

[21] SMO document collection - DocSur_ERJU_A6Z00064243558/PM1/000/-

[22] D10.1 and D10.3 – Annex GTSD – Reference to internal document and deliveries, FP2-WP10-A-GTD-028-01

Interne

[23] Internal/auditable GSTD document documented in [22], Company number 68088834-305~15870~9950

[24] Internal/auditable GSTD document documented in [22], Company number 68088841-305~15870~9950

[25] Test Report – ADM Tests for R2DATO Internal/auditable GSTD document documented in [22], Company number 68088838-279~15870~9950.

[26] Test-case specification – APM tests for R2DATO Internal/auditable GSTD document documented in [22], 68088837-206~15870~9950

[27] D5.1 - Documentation of use cases for automating functions - FP2-T5_1-D-NSR-124-06, Annex 4 - Use cases for Automating Functions

[28] Documentation of use cases for Perception system - FP2-T5_2-D-FTS-040-05

[29] X2RAIL-4 project, Deliverable D5.1 – SRS ATO up to GoA3/4 version 1.0.0 (LA_GoA34_S2R_1.0.0_20231214.docx) - X2R4-WP05-D-ALS-010-013

[30] X2RAIL-4 project, Deliverable D5.1 – Dictionary (LA_GoA34_S2R_1.0.0_20231214.xlsx) – X2R4-WP05-D-ALS-010-013

[31] X2RAIL-4 project, Deliverable D5.1 – Capella model (LA_GoA34_S2R_1.0.0_20231214_Capella_5.0.0.zip) – X2R4-WP05-D-ALS-010-013

Interne