COAT

CCS onboard application platform for trackside related functions

SYSTEM DOCUMENTATION

COAT Demonstrator

Version 2.0, released 29.11.2019, status: draft

Disclaimer

This document is a DRAFT version which is still under construction. Its content may change in the ongoing concept phase of smartrail 4.0. The document is not completely verified and is not finalized. The document is published to enable an open discussion of the ongoing work of the smartrail 4.0 program.

Links and references inside of this document may refer to other documents inside of the program smartrail 4.0, that may not be published at this stage.

Table of Contents

1 Introduction 7

1.1 Context 7

1.2 Target Audience, Goal and Structure of the Document 7

2 System Architecture 8

2.1 Phase A – Simulation 8

2.2 Phase B – Laboratory Testing 9

2.3 Phase C – Field Testing 10

2.4 Phase D – Laboratory Testing 11

3 Hardware Components and Interfaces (Physical) 12

3.1 Onboard Unit (OBU) 12

3.2 Driver Machine Interface (DMI) 14

3.3 Juridical Recording Unit (JRU) 15

3.4 Train Interface Unit (TIU) 16

3.5 Wheel Speed Sensor 16

3.6 Euro Radio 16

3.7 Balise Transmission Module (BTM) and Antenna 17

3.7.1 BTM Specification 19

3.7.2 BTM Antenna Specification 20

4 Software 21

4.1 Security & Safety 21

4.2 DMI 21

4.3 BTM 21

4.4 JRU 21

4.5 Odometer 22

4.6 TIU 22

4.7 Test Bench 22

4.7.1 Train Simulator 22

4.7.2 Track Editor 23

4.7.3 Scenario Editor 23

4.7.4 Scenario Controller 24

4.7.5 Database Manager 25

4.7.6 Scenario Analyzer 25

4.7.7 Traffic Simulator 26

4.7.8 RBC Message Simulator 27

4.8 OBU 28

4.8.1 OBU Layered Software Architecture 28

4.8.2 OpenETCS Kernel (SUBSET-26) 28
4.8.3 Odometer Interface Manager (FFFIS 97E2675B) 29
4.8.4 Balise Interface Manager (SUBSET-36) 29
4.8.5 Train Interface Manager (SUBSET-034 & SUBSET-119) 29
4.8.6 Recorder Interface Manager (SUBSET-27) 29
4.8.7 DMI Interface Manager (SUBSET-121) 29
4.8.8 Euro Radio Interface Manager 29
4.8.9 Bitwalker 30
4.8.10 Application Manager 30
4.8.11 Runtime Environment 30
4.8.12 Drivers 30
4.9 Phase A 31
4.9.1 Components 32
4.10 Phase B 33
4.10.1 Components 33
4.11 Phase C 34
4.11.1 Components 34
4.12 Phase D 34
4.12.1 Components 35
4.13 Summary 36

5 Test Use-Cases 36

6 Conclusions 36

7 List of Suppliers 36

Revision History

Table 1

Date	Version	Description	Author(s)
21.06.2019	1.0	Initial edition	COAT Team
29.11.2019	2.0	• Reworked based on some remarks. • Added the phase D.	GD / CG

Reference documents

Title	Reference	Version
COAT Demonstrator – TESTS OBJECTIVES	TEST_002	v 1.2
20190731_OR-D3-1907-01 BTM Manual	BTM_001	v 1.0
ODOMETER FFFIS	ODO_005	v 5.0
M72 - Motion Counter	M72_001	v 1.0
TIU Interface Specification for SBB COAT demonstrator	TIU_001	v 1.0
Phase B (B1) General specification	SPB_001	v 4.3
ORD Software Requirements Specification	5.2423.018/03V10	v A02

Table of Abbreviations and Acronyms

Abbrevia-tion	Definition
BCS	Balise Communication Simulator
BTM	Balise Transmission Module
CCS	Command and Control System
COAT	CCS onboard application platform for trackside related functions
DMI	Driver Machine Interface
ERA	European Railway Agency (European Union Agency for Railways)
ERTMS	European Rail Traffic Management System
ETCS	European Train Control System
EVC	European Vital Computer
FFFIS	Form Fit Function Interface Specification
ID	Identification
IF	Interface
JRU	Juridical Recording Unit
OBU	Onboard Unit
OCORA	Open CCS On-Board Reference Architecture
ODO	Odometer
PoC	Proof of Concept
RBC	Radio Block Center
RIM	Radio Interface Module
TCMS	Train Control and Monitoring System
TIU	Train Interface Unit
ZUB	Zugbeeinflussung (Train Control System used in Switzerland)

1 Introduction

1.1 Context

This document is about the program phase “Demonstrator” of smartrail’s 4.0 program COAT. COAT stands for “CCS onboard application platform for trackside related functions”, hence hereunder we focus on the synergy potential, addressing the common needs of the onboard functions.

This demonstrator is a rapid prototype of a CCS onboard solution, based on off the shelf software and hardware components. It will serve as first applicant for a new modular certification process. This certification concept will be published separately.

The demonstrator focuses on two main aspects:

- PoC to apply a new modular homologation approach
- PoC if the draft CCS onboard architecture can be realized with already available products to ensure upgradeability

Although the demonstrator has ETCS functionality, it has no focus on a specific functionality scope nor is it related to the functional scope of smartrail4.0.

1.2 Target Audience, Goal and Structure of the Document

The primary target audience is meant to be smartrail 4.0 internal staff in general and COAT members specifically. In addition, the Swiss and European authorities will be served related to the vehicle test setup / adaption and temporary permit and the first application of a new modular certification process. Members of the European initiative OCORA will be served with the conclusions. For industry representatives it can be a source of information for future product developments.

This paper provides the content, the system architecture, the hardware components, emulated devices, the software components, the interfaces and a first set of test use-cases. At a later stage the conclusions will be added.

2 System Architecture

The COAT demonstrator project is implemented in 4 phases. The following chapters provide a top-level view of the architecture for each phase. In the architectural diagrams the following notation is used:

- Blue rectangles: hardware elements
- Green rectangles: software elements (function groups or functions)
- Red rectangles: interfaces

2.1 Phase A – Simulation

In phase A, the COAT demonstration system is prepared. For that purpose a PC and an external DMI is used. A simulation software that includes an open ETCS kernel is developed/configured to run on the PC, communicating over Ethernet with the ETCS DMI software.

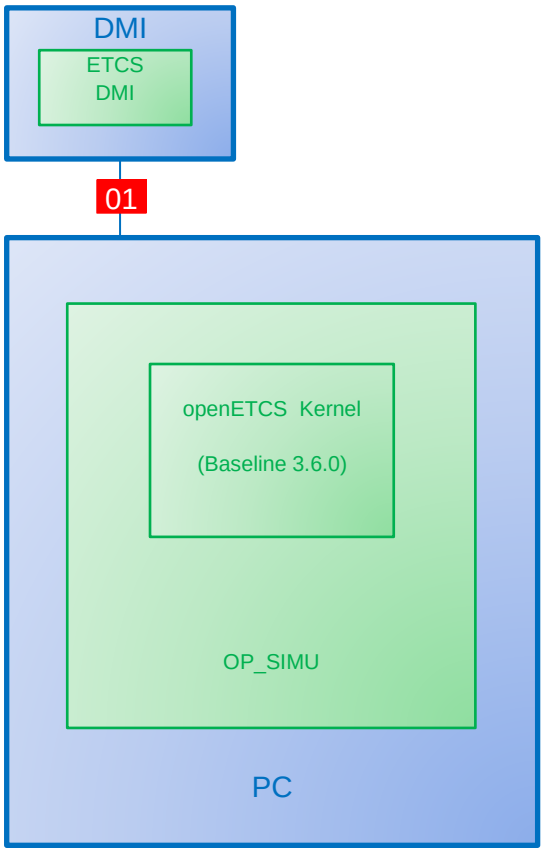


Figure 1 System Architecture – Simulation Phase A

2.2 Phase B – Laboratory Testing

The purpose of the Phase B is to reduce the integration risks on the train by using as much as possible the same components which will be used on the train, and with the use of simulations. The full description of this phase is found in the document [SPB_001].

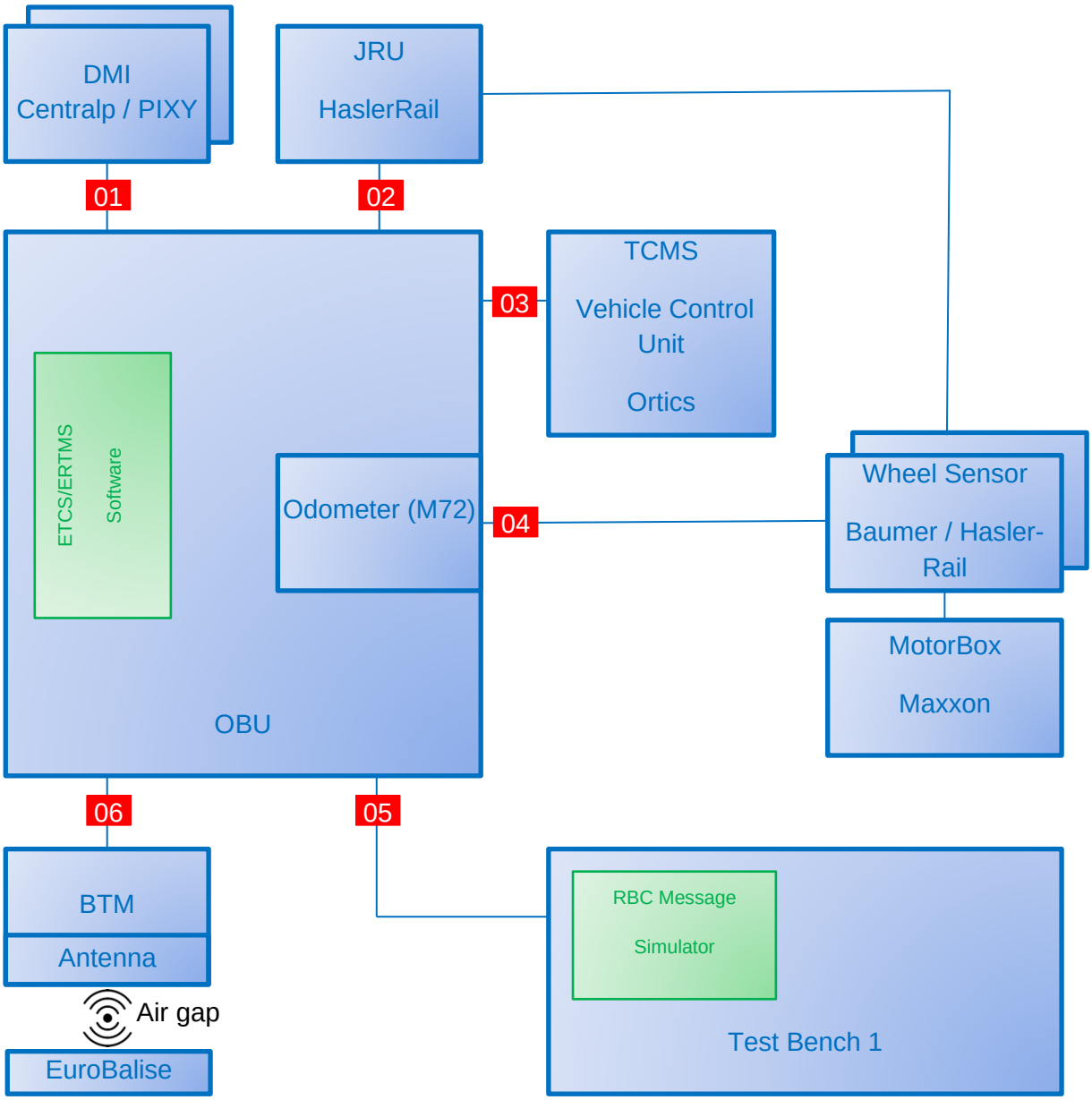


Figure 2 System Architecture – Laboratory Testing Phase B

2.3 Phase C – Field Testing

In Phase C, the COAT demonstration system is integrated in a Tmf232 vehicle for field testing purposes. The connectivity to the RBC continues to be simulated while all other emulations/simulations are replaced by the real vehicle equipment.

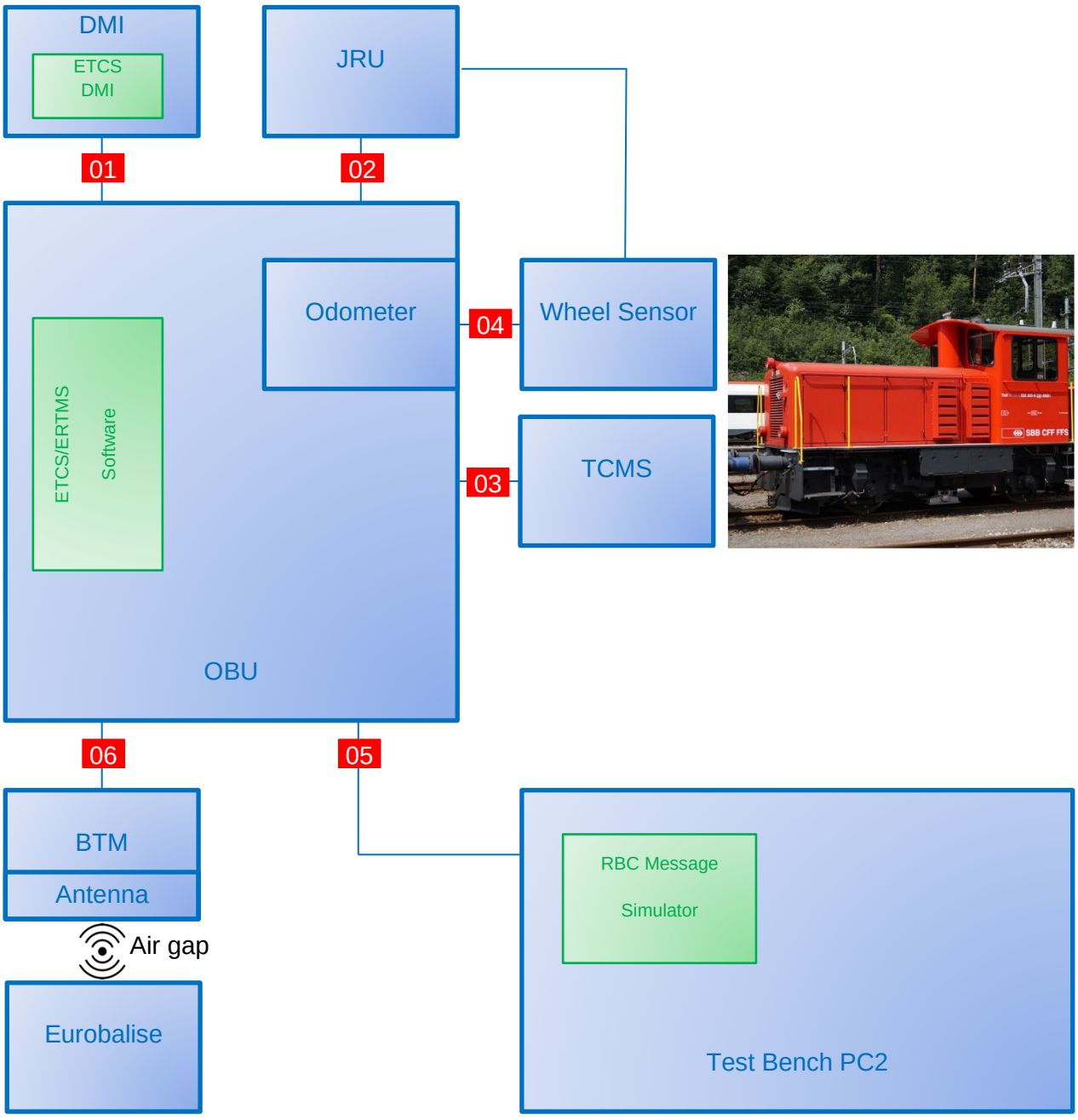


Figure 3 System Architecture – Field Testing Phase C

2.4 Phase D – Laboratory Testing

In Phase D, the COAT demonstration system is setup in the laboratory for more automated testing. For that purpose an OBU hardware, a DMI, a JRU and a BTM including antenna are integrated. To test the system, a test bench is used, consisting of 2 PCs, a Train Interface Unit (TIU) emulator, a wheel sensor emulator and a Balise Communication Simulator (BCS).

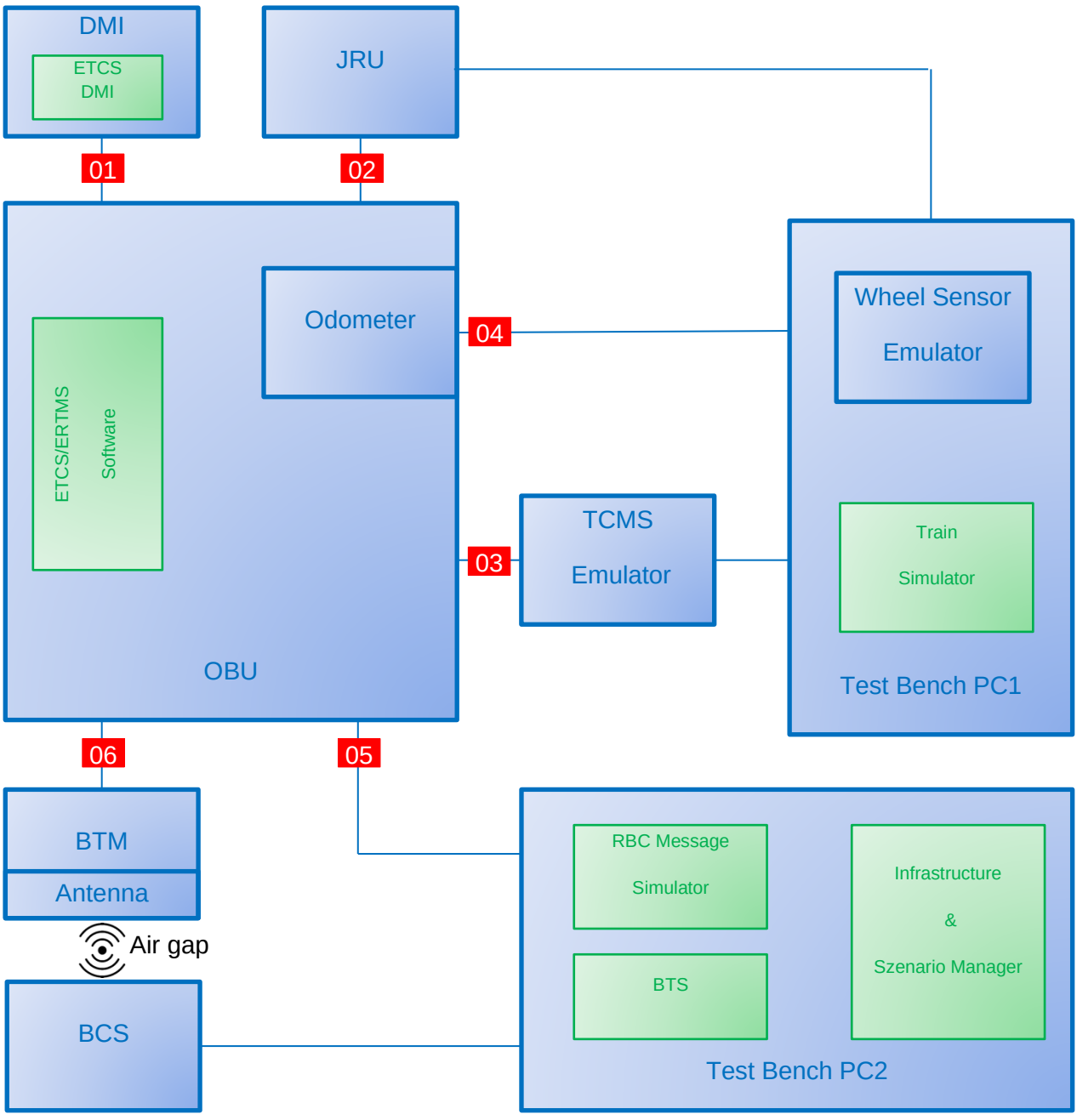


Figure 4 System Architecture – Laboratory Testing Phase D

3 Hardware Components and Interfaces (Physical)

3.1 Onboard Unit (OBU)

In the context of the ETCS, the OBU is called EVC. It is the core of the automatic train protection system and is the unit interacting with vital train functions, such as activating the brakes and cutting-off the traction. Furthermore, it handles the wheel sensors signals or the GSM-R data reception. For the COAT demonstrator the term OBU is used, since this hardware platform is not limited to ETCS functionality only.

The OBU chosen for the COAT demonstrator is a SIL4 hardware platform provided by MEN, powerful enough to support real-time communication, voting and to flawlessly process at least all the applications required for an ETCS.

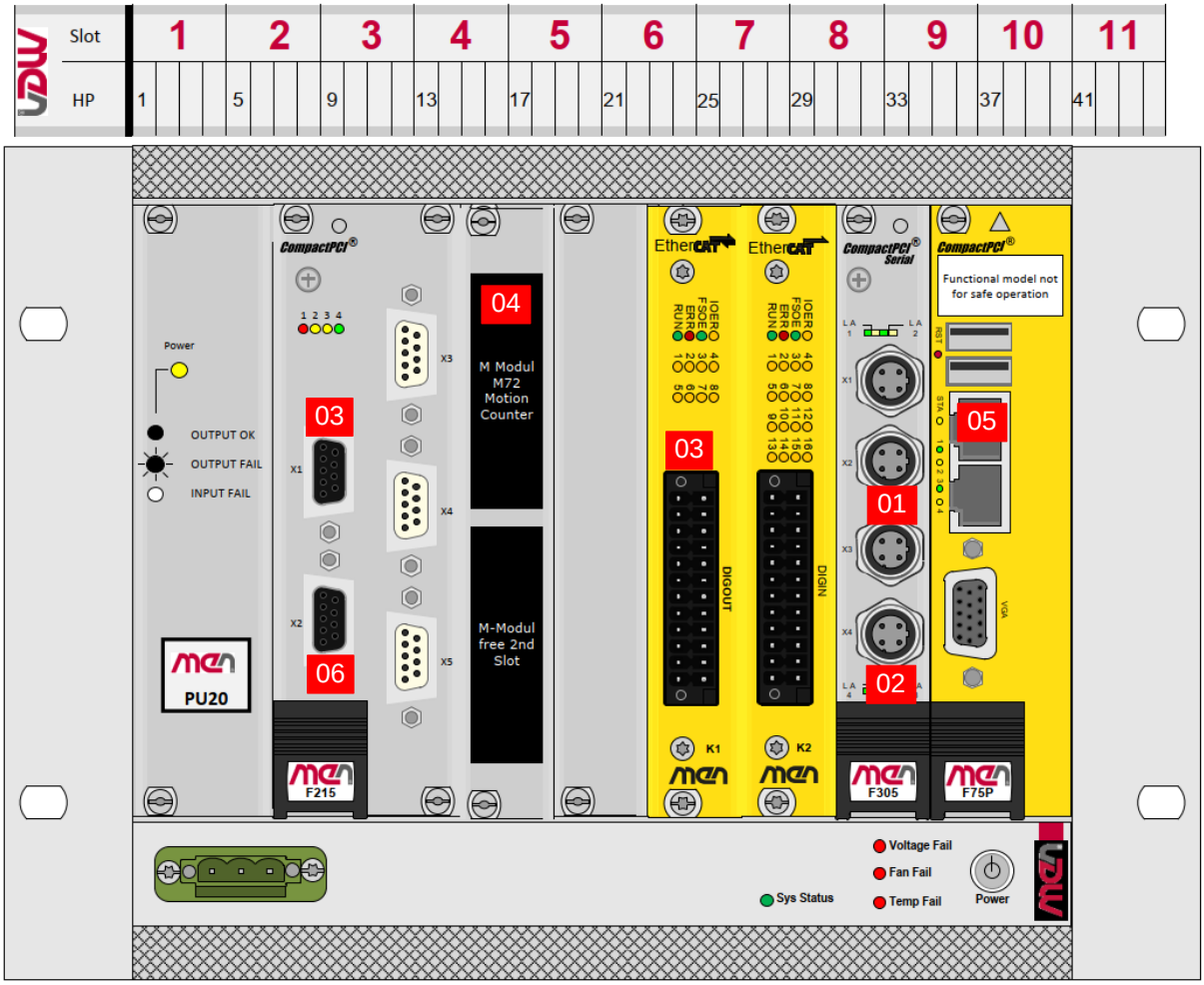


Figure 5 Onboard Unit (incl. identification of used Interfaces)

Feature	Specification
Model Name	19MH50CB0
Supplier	MEN (*https://www.men.de/products/*)
Manufacturer	MEN (*https://www.men.de/products/*)
General Characteristics	Modular design, built-to-order configuration

Feature	Specification
Interfaces	See text below
CPU / RAM	CPCI 3U Board; 3x Intel Atom E680T (1.6 GHz), 2x 512 MB, -40°C to +85°C, air-cooled, conformal coating
RAM	1 GB DDR2 DRAM
Safety	▪ Certifiable up to SIL 4 according to EN 50129/EN 50128 and up to SIL 3 according to IEC 61508, depending on I/O board configuration ▪ Hazard rate for safety functions <= 1E-9 / h ▪ System maintains safe state after a failure ▪ EN 60950-1: Class I equipment ▪ UL 94 V-0 ▪ EN 45545-2, hazard level HL3 (19MH50CB0 barebone configuration)

For security and safety purpose, the OBU hardware module has a secure boot function to protect from malware. For the implementation of the COAT demonstration system, the following interfaces are used.

IF ID	Description
01	▪ 1 x Ethernet for the connection to the DMI (similar to SUBSET-121)
02	▪ 1 x Ethernet for the connection to the JRU (SUBSET-027). Refer to chapter 4.4 for more details.
03	▪ 1 x CAN for the connection to the TCMS (SUBSET-034, SUBSET-119) ▪ Additional discrete IO connections to the demonstrator vehicle (Emergency brake, Traction Cut Off) are foreseen (SUBSET-034, SUBSET-119).
04	▪ 1 x discrete wheel sensor input. The wave signal from the wheel sensor is converted in speed and direction by the OBU odometer functionality. This information is forwarded to the kernel application with the following protocol [M72_001] Refer to chapter 4.5 for more details.
05	▪ 1 x Ethernet Interface for the connection to the Euro Radio and RBC Emulator (SUBSET-037).
06	▪ 1 x CAN for the connection to the BTM. Refer to chapter 4.3 of the document [BTM_001]for more details.

3.2 Driver Machine Interface (DMI)

The DMI is the user interface between the driver and the ETCS system. For the COAT demonstrator, a unit from CENTRALALP is used, hosting the ETCS DMI application software.

Figure 6 Driver Machine Interface

Feature	Specification
Model Name	e-Vision+ DMI 10.4”
Supplier	ClearSy (*https://www.clearsy.com/en/*)
Manufacturer	CENTRALALP (*https://www.centralp.fr/en/*)
Touch Screen	10.4” XGA LEDs, capacitive touch
Front panel hard keyboard	Yes
CPU	Imx6 DUAL CORTE 800MHz
RAM	2Go DDR3
Mass Storage	eMMC 4GB
Power supply	1 (M12 connectors)
Input Voltage	24 – 110VDC
Ethernet	2 (M12 connectors) 10BaseT and 100BaseT Ethernet
USB	1 minimum (M12 connector)
Audio	1 (D-Sub 9 Male connector)
Debug	1 (D-Sub 9 Female connector)
OS	Linux

3.3 Juridical Recording Unit (JRU)

The JRU provides ‘black box’ functions, i.e. it stores the most important data and variables from train journeys, allowing for later analysis. For the COAT demonstrator a unit from HaslerRail is used.

Figure 7 Juridical Recording Unit

Feature	Specification
Model Name	TELOC®3000
Supplier	HaslerRail (*https://www.haslerrail.com/*)
Manufacturer	HaslerRail (*https://www.haslerrail.com/*)
Case	Small
Memory	8GB
Power Supply	24 – 110VDC
Service Interface	1x RJ45, 1x USB, 1x M12 Ethernet
Axle Encoder Interface	1x TACHA Card
Environment Protection	IP20
Interfaces	MVB, CAN, Profibus, RS485, Ethernet, USB Type A
ETCS Functionality	Baseline 2 & 3
Standards	EN 50155, IEEE 1482.1

The HaslerRail TELOC®3000 data recording system can provide certifications up to SIL 2 and 4 for specific functions. These however are not used in our context.

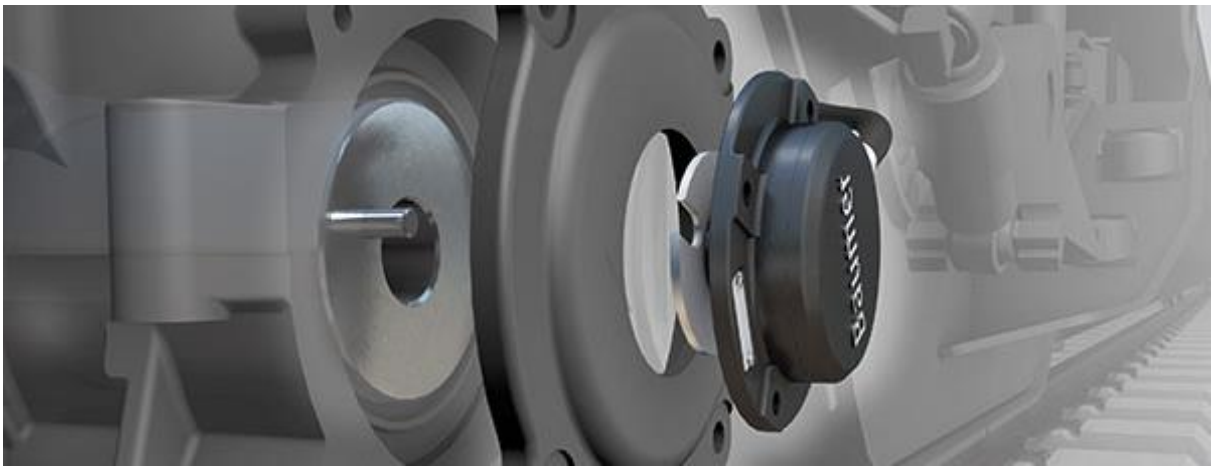
The system can acquire data from different sensors and from most third-party systems via different interfaces. It can provide speed calculation, data recording, data processing, speed indication and signal transmission.

3.4 Train Interface Unit (TIU)

During phases A and D of the COAT demonstrator project, the TIU is simulated/emulated. The TCMS system on the demonstrator vehicle has a TIU, based on CAN bus.

3.5 Wheel Speed Sensor

The vehicle used for the COAT demonstrator has BMIV axle encoder provided by Baumer.



During phases A and D of the COAT demonstrator project, the wheel speed sensor signal is simulated/emulated.

Feature	Specification
Model Name	BMIV 58L1624K/406045
Supplier	Baumer (*https://www.baumer.com/ch/en/*)
Manufacturer	Baumer (*https://www.baumer.com/ch/en/*)
SBB Part Number	625-56-350

3.6 Euro Radio

The RBC (Radio Block Centre) and data connection to RBC will be simulated/emulated by the testbench in all phases of the COAT demonstrator project. Refer to chapter 4.7.8 for details.

3.7 Balise Transmission Module (BTM) and Antenna

The BTM and antenna are two separate components, designed to read data from Eurobalises. They detect the balise center, process the data received from the balise, build a telegram and forward the information to the OBU.

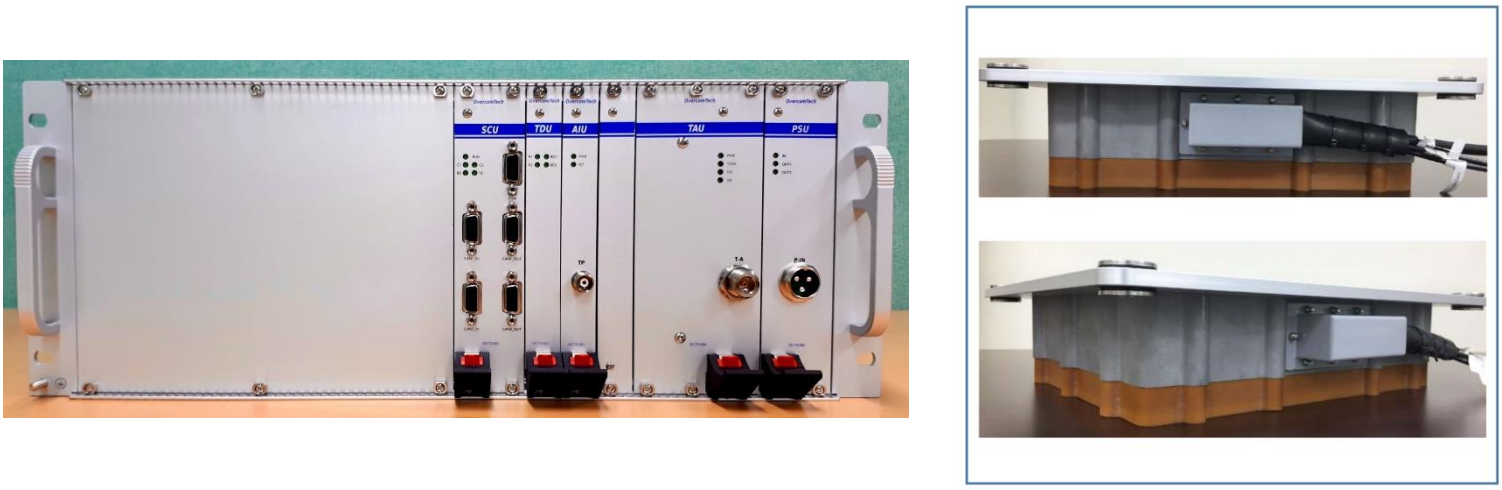


Figure 8 BTM Module (left) and Antenna (right)

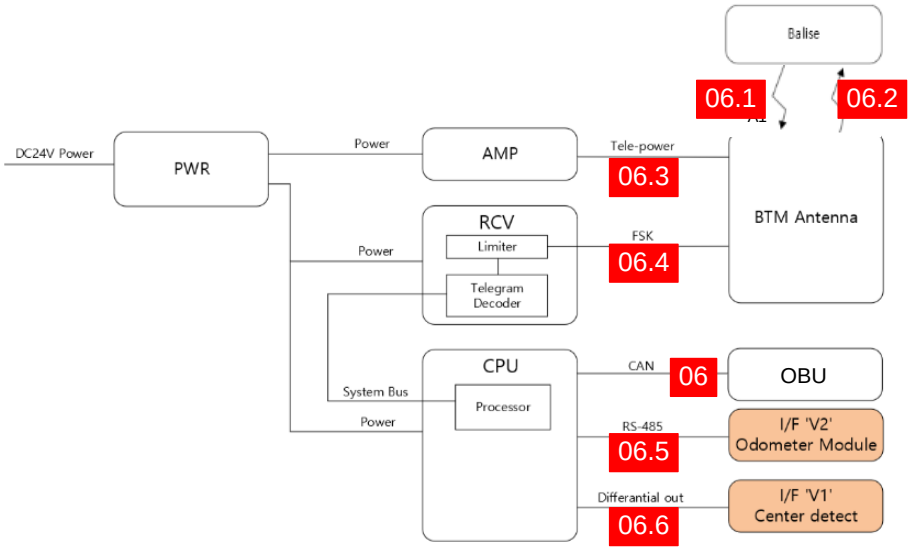


Figure 9 Logical Component Configuration & Interfaces

The interfaces of BTM and antenna are as follows:

- Interface 06 : CAN interface for transmitting user telegrams to OBU (refer also to chapter 3.1).
- Interface 06.1 : for transmitting up-link Eurobalise telegrams from the Eurobalise to the antenna.
- Interface 06.2 : for transmitting the required power from the antenna to the Eurobalise.
- Interface 06.3 : 'Tele-power' N-type interface for transmitting tele-power to antenna.

- Interface 06.4 : ‘FSK’ TNC interface for receiving up-link Eurobalise telegrams from antenna.
- Interface 06.5 : RS-485 interface ‘V2’ for testing the Eurobalise transmission system.
- Interface 06.6 : ‘Differential out’ interface ‘V1’ for testing the Eurobalise transmission system.

3.7.1 BTM Specification

Feature	Specification
Supplier	Asystems Co. (*http://www.asystems.co.kr*)
Manufacturer	OvercomTech (*http://www.ovctech.com*)
IF BTM – Antenna	2 x high frequency connectors with 50 Ω impedance

The BTM is designed as standalone fail-safe equipment.

- Supported Balise Types:
  - Standard Balise
  - Reduced Balise
  - Long/short telegram

- Compliant with CENELEC EN 50155, 50129, and 50128
- Compliant with EUROBALISE specification
- The BTM module is made up of the following sub-modules:
  - Tele-powering transmission unit
  - Telegram reception and processing unit
  - OBU Interface unit
  - Power supply unit

The BTM has the following characteristics:

Characteristics			Value	limitation	Description
BTM	Supplied Voltage		DC18V ~ DC36V	Compliant EN50155	Typical 24VDC
	Consumption Power		50W(standby 12W)	Max. 60W	
	Signal	Transmission	27.095Mhz	Compliant Subset 036	High Speed Power amplifier
		Reception	4.234Mhz		564.48kbit/s, FSK Demodulator
		Size	261.2x222.2x188.19		WxHxD mm
		Weight	6.7kg		±0.5kg

Figure 10 BTM Characteristics

During phase D of the COAT demonstrator, the Eurobalise is emulated. This means that no physical balise provides the data, but the BCS allowing a certain degree of test automation.

3.7.2 BTM Antenna Specification

Feature	Specification
Supplier	Asystems Co. (*http://www.asystems.co.kr*)
Manufacturer	OvercomTech (*http://www.ovctech.com*)
IF BTM – Antenna	2 x high frequency connectors with 50 Ω impedance

The BTM antenna is designed for tele-powering and receiving telegram signals.

- Compliant with CENELEC EN 50155, 50129, and 50128.
- Compliant with EUROBALISE specification
- The Antenna module is made up of the following sub-components:
  - 27 MHz Tele-powering Antenna unit
  - 4.2 MHz receiver Antenna unit
  - Impedance Matching elements
  - Monitoring supervision Logic
  - Mechanical connection interface

The BTM antenna has the following characteristics:

Characteristics			Value	limitation
BTM Antenna	Mechanical	Size(mm)	448*127*314	-
		Weight(Kg)	11.2kg	±0.1kg
	Environments	Temperature	-40°C ~ +80°C	-
		Seal grade	IP67	-

Figure 11 BTM Antenna Characteristics

4 Software

4.1 Security & Safety

The Demonstrator is a non-safe application, meaning the software safety standard CELENEC EN 50128 has not been taken in consideration in the development of the applications for the demonstrator. Nevertheless, the SIL4 platform of MEN is certified to EN 50126, EN 50128 and EN 50129.

4.2 DMI

The Driver Machine Interface (DMI) software provides a cab display allowing the driver to input information into the ERTMS/ETCS system; it allows driver identity and train data, to be entered, and the display of driving information, including speed, planning area, ETCS level, and mode. The DMI also prompts driver actions: selection of driving mode, confirmations, and acknowledgements. It satisfies ERA DMI specifications (ERA_015560) and supports ERA's recommendations on area design. ETCS levels 0 to 2 are all supported, as are all technical modes requiring an open driver's desk including Standby, Unfitted, Shunting, Staff Responsible, Full Supervision, On Sight, Trip, Post Trip, Reversing, Non-Leading, Limited Supervision, and National Systems. Automatic testing and self-test functionality are built in. In addition to the ERA specification some national system layouts are also included: ATB EG, KVB/TVM, PZB, AWS/TPWS, ATC-2, and ZUB. Regarding the protocol, the following is valid:

- Physical Interface: Ethernet
- Protocols: Clearsy proprietary
- Message: Clearsy proprietary

4.3 BTM

The BTM software complies fully to the FFFIS defined in the SUBSET-36 for the communication to the Eurobalise. The communication protocol between the OBU and the BTM is defined in the User Manual [BTM_001] of the BTM provider.

4.4 JRU

The software of the JRU device enables to periodically record all the information provided by the OBU. There are other tools for the download and analysis of these information. The software fully implements the FFFIS of the SUBSET-27. The Communication protocol between JRU and OBU is described in 5.2423.018/03V10. No safety layer will be implemented.

- Physical Interface: Ethernet
- Protocol: HaslerRail and Clearsy proprietary
- Message: HaslerRail and Clearsy proprietary

4.5 Odometer

The Odometers von Baumer and HaslerRail will be connected to the M72 motion board [M72_001] provided by MEN. This M72 M-Module™ is a timer, counter and comparator with four 32-bit counter units. The four counters work independently in several modes, e.g. counting decoder signals or measuring frequencies. Each counter can be loaded with a pre-set value and can generate several events. Events can set or reset output signals, which are optically isolated. The M72 can also be used to generate timed signals. In addition, it features line break detection. The sensor signals are optically isolated and can be used as TTL, RS422 or 24V inputs. The Form Fit Functional Interface Specification (FFFIS) between the ERTMS Kernel and the Odometer Unit is described in the document [ODO_005].

4.6 TIU

The Train Interface Unit software implements the functionalities described in the SUBSET-119 and -034. Its purpose is to enable the control of the main functions of the train, such as:

- Braking system of the train, both service and emergency brake
- Train control, to command the change of traction, raising and lowering of pantograph on the roof and the air tightness (the latter out of scope for the demonstrator project).
- Engine control, to cut the traction power when either the service brake or emergency brake is activated.
- Cab status includes determining the position and check the status of the cab desk (open or closed).

All those functions are not mandatory for the Demonstrator project. It is the reason why the functional scope has been simplified. The resulting interface description is described in the document [TIU_001].

The CANopen standard will be used between the OBU and the TIU.

4.7 Test Bench

The Test Bench enables a laboratory simulation of components which are not available as long as the OBU is not deployed on a vehicle. It also provides the RBC functionality and enables the definition and realization of test scenarios in the laboratory and on the train (RBC). The OBU Test Bench is compliant with the reference architecture defined in SUBSET 094.

- Using the **off-line tool**, users can also create testing information including train parameters, speed profile, messages based on ETCS baseline 2 or baseline 3.
- The **on-line tool** is used to define and control a **test environment** which is connected to the on-board system to be tested using standard interfaces.

4.7.1 Train Simulator

The test environment in the OBU Test Bench is made up of several modules working together:

- **scenario controller** to start, manage and stop scenarios
- **driver simulator** to automatically enter data on the DMI and perform actions on the desk

- **train motion simulator** to perform the train dynamics
- **speed sensor simulator** to deliver the odometry information to the OBU
- **TIU simulator** to enable and stimulate the interfaces between OBU and TCMS
- modules to simulate the transmissions by balise, loop, radio, and national signaling equipment
- modules to record and review all logged information.

The train simulator supports all ERTMS/ETCS levels can be fully customized by the user and simulate train dynamics in baseline 1, 2 or 3. Train objects can be driven manually and start from any location on the track, and in automatic mode they can run according to the timetable and the driving information and data received over the balise and radio interfaces.

4.7.2 Track Editor

The Track Editor allows defining the track topology used in the rest of the system. To ensure the usage of consistent and accurate topology data, the Track Editor is delivered with an XML based importation function compliant with SUBSET-112

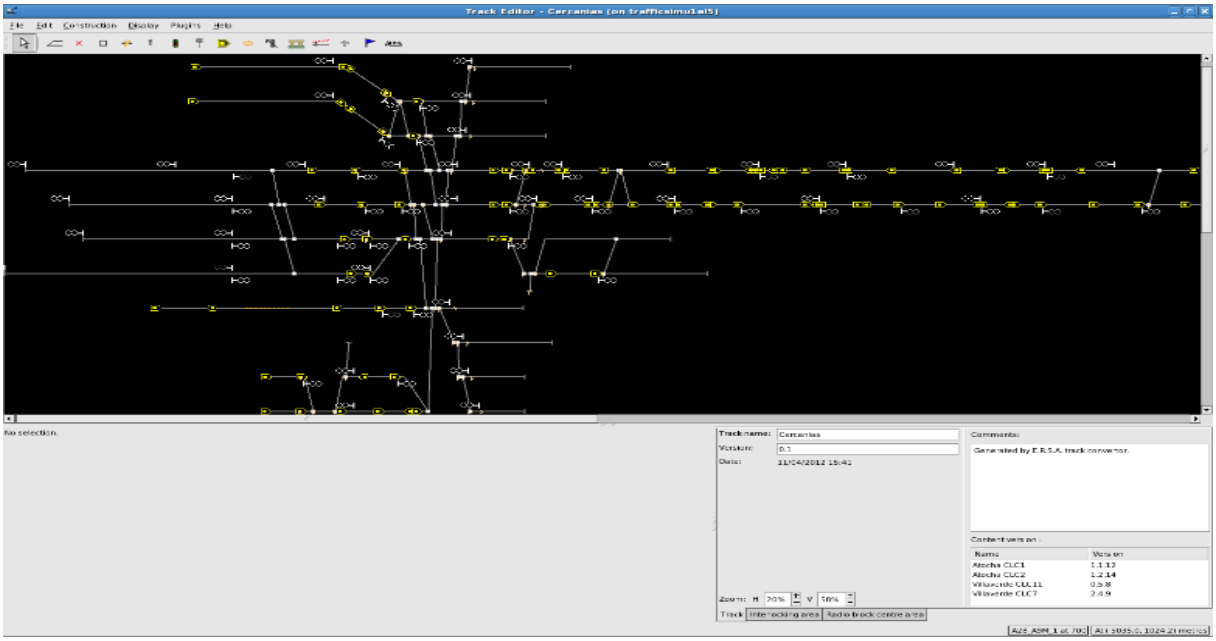


Figure 12 Track Editor User Interface

4.7.3 Scenario Editor

The Scenario Editor allows the creation and edition of scenarios. A scenario is based on a specific track and consists mainly of:

- specifying the initial train location and orientation;
- specifying the train equipment and configuration;
- specifying the ETCS telegrams and messages;
- configuring the signal aspects and point positions.

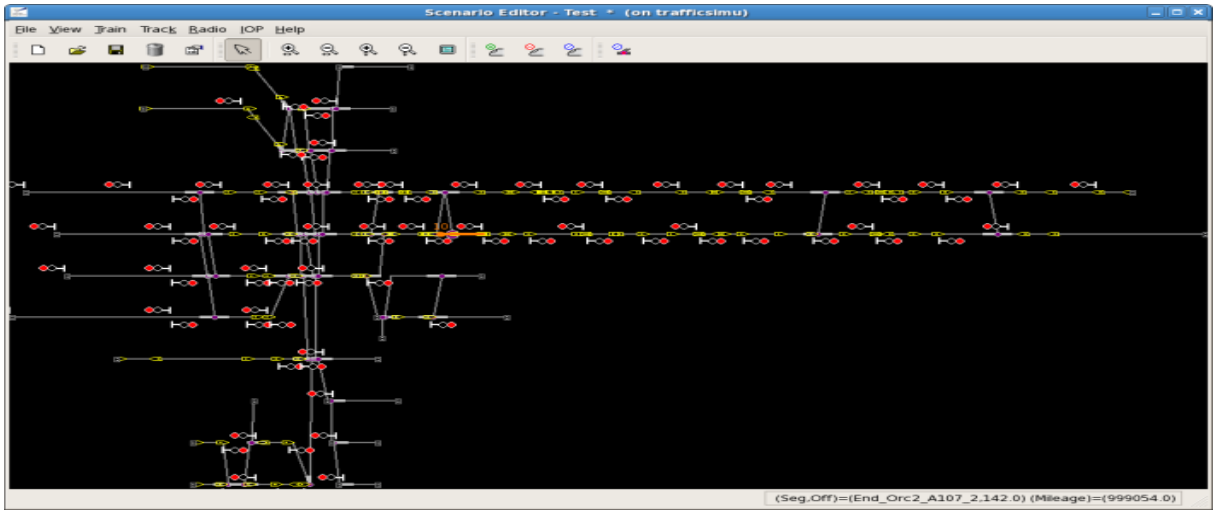


Figure 13 Scenario Editor User Interface

4.7.4 Scenario Controller

Once a scenario is defined, the Scenario Controller is used to execute it as a test run. This module is in charge of starting and controlling the various other online modules (Route Map Controller, …) which depend on the product in which it is included.

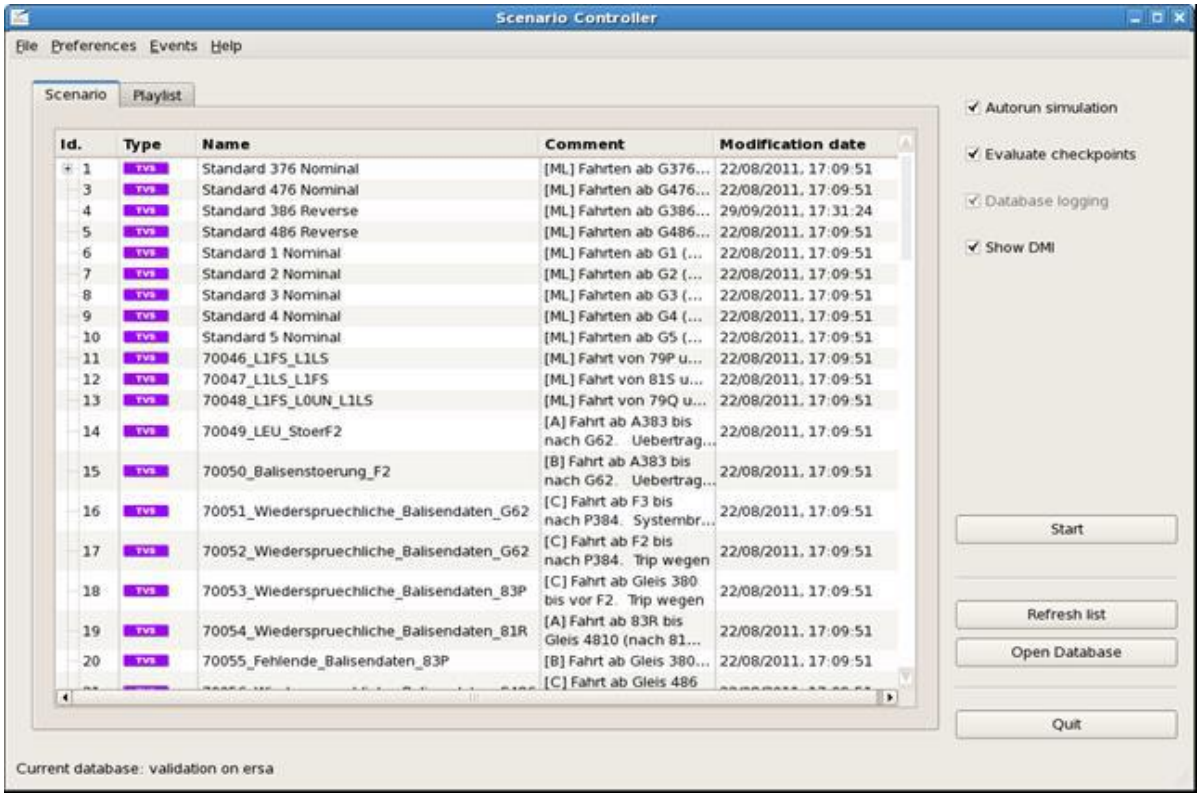


Figure 14 Scenario Controller User Interface

4.7.5 Database Manager

The Data Managed by ClearSy simulators and test benches are stored in MySQL databases. The Database Manager allows performing basic database operations like creation, copy, deletion, backup, restoration, and also takes care of upgrading the format/content of existing databases in case of format/content evolution.

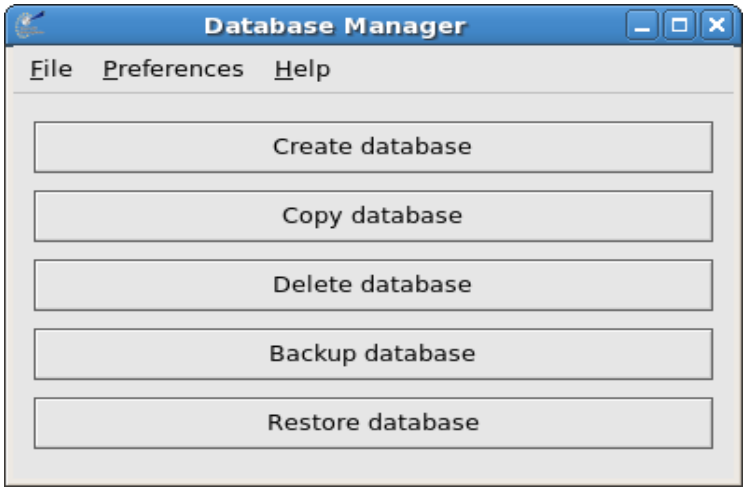


Figure 15 Database Manager user interface

4.7.6 Scenario Analyzer

The Scenario Analyzer is an offline tool that allows reviewing the data logged during the test run like:

- ETCS telegrams/messages exchanged with the train
- Odometry information
- TCMS information
- Braking curve information
- It can also be used to open JRU data (SUBSET-027) produced by industrial or simulated EVCs.

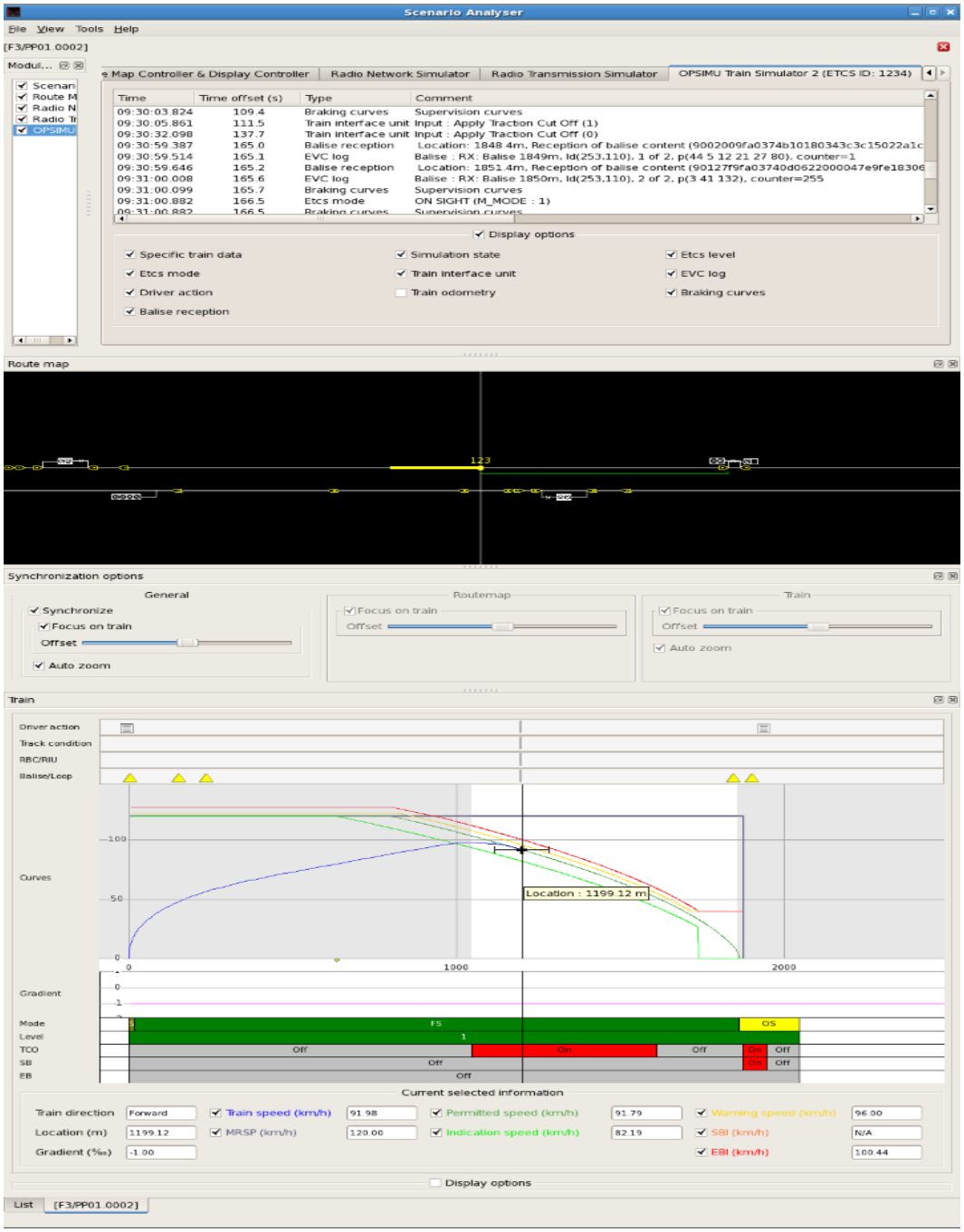


Figure 16 Scenario Analyzer Graphical User Interface

4.7.7 Traffic Simulator

The ERTMS/ETCS Traffic Simulator is a real-time system which can be used for evaluating line capacity and improvements, validating RBCs, assessing conflict detection/resolution systems, signaler training and replicating real life events.

The traffic simulation is managed by a **Scenario Controller** which initiates or stops the simulation and allows users to inject events or failures in track or train.

A **Route Map Manage**r shows the track topology and all train movements and allows users to show or hide other features: balises, track circuit status, track profiles, boards and signals.

The **Interlocking** allows users to set and lock routes; and provides block condition status required by the Radio Block Center module to send movement authorities to the trains.

The **Radio Block Centre (RBC Message Simulator)** manages connection and disconnection to trains, issues movement authorities TSRs and text messages; receives position reports, sends and revokes emergency stops; and displays the new position of trains as soon as a report is received. RBC can also generate diagrams showing actual and predicted positions of all trains in the simulation.

4.7.8 RBC Message Simulator

The Euro Radio simulator is in charge of sending predefined radio messages to the OBU and receiving radio messages from the OBU Kernel. It manages connection and disconnection to trains, issues movement authorities TSRs and text messages; receives position reports, sends and revokes emergency stops; and displays the new position of trains as soon as a report is received. RBC can also generate diagrams showing actual and predicted positions of all trains in the simulation.

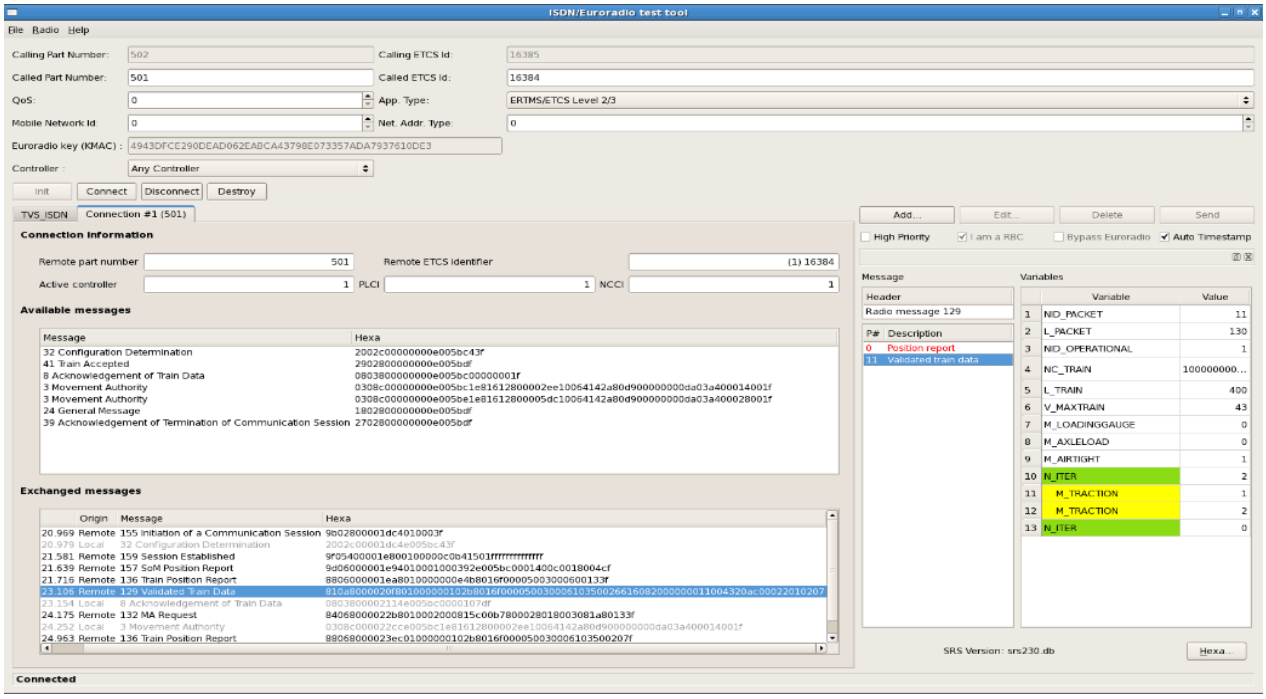


Figure 17 Euro Radio Test Tool

4.8 OBU

4.8.1 OBU Layered Software Architecture

The following graphic describes the software architecture of the OBU software.

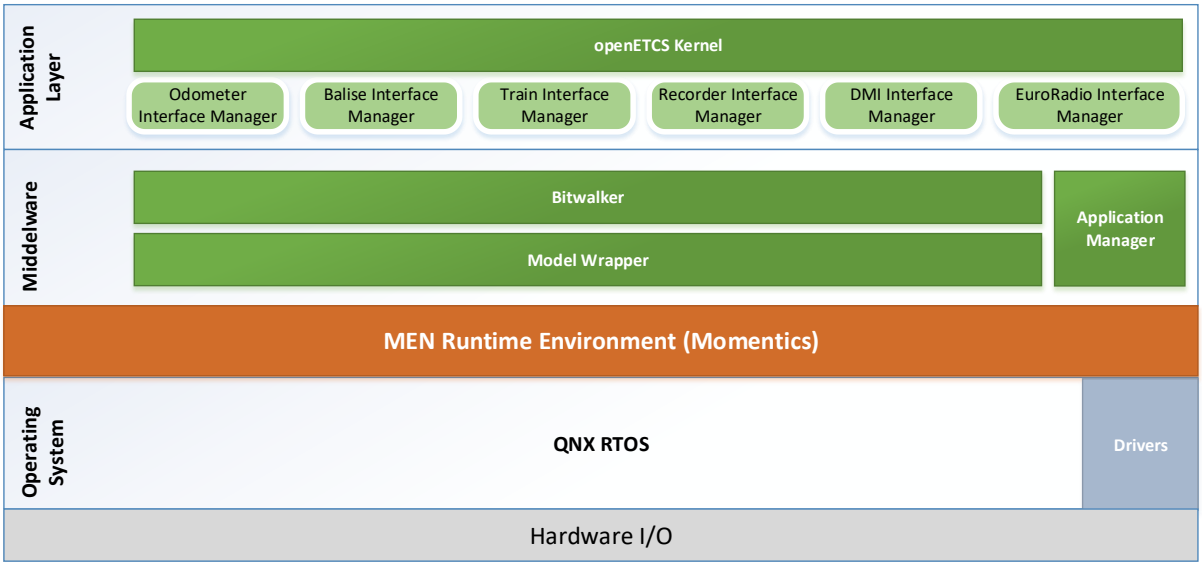


Figure 18 Architecture of the OBU Software

4.8.2 OpenETCS Kernel (SUBSET-26)

The openETCS Kernel as mainly specified in UNISIG SUBSET-026 version 3.3.0 is fundamentally, the center of all the control process and protection of the train movement. It performs the following main tasks:

it checks the state of the system and performs all mode and level transitions;

- it validates the data entered by the driver;
- it calculates the train supervision curves;
- it reads the balise, loop and radio telegram and processes the data;
- it validates the RBC-Emulator messages and processes the data;
- it transmits the train location and other information to the RBC-Emulator;
- it manages the movement authorities from the RBC-Emulator, balises or loops;
- it calculates the most restrictive speed limits for the current train location;
- it supervises the train movements (including the protection against undesirable movement)
- and triggers warnings and interventions;
- it records all events occurring in the system.

4.8.3 Odometer Interface Manager (FFFIS 97E2675B)

This module collects odometer information (position, speed, acceleration) from the odometer device of MEN and transmit them to the openETCS Kernel.

4.8.4 Balise Interface Manager (SUBSET-36)

The standard communication protocol is defined by OvercomTech.

It performs the following tasks:

- It receives and decodes the balise and loop message
- It manages the balise groups
- It manages the balise linking
- It manages the track condition about big metal masses

4.8.5 Train Interface Manager (SUBSET-034 & SUBSET-119)

The communication protocol will correspond to the one used for the TCMS of the train

It performs the following tasks:

- It receives and processes the odometry data to update the internal location data
- It updates the OBU commands on the TIU
- It gets the train statuses from the TIU

4.8.6 Recorder Interface Manager (SUBSET-27)

It manages the recording of juridical data. The communication protocol is a simple standard defined by ClearSy (ERSA) in collaboration with HaslerRail.

4.8.7 DMI Interface Manager (SUBSET-121)

It performs following tasks:

- It manages the communication with the DMI
- It sends relevant data for display
- It manages the train data entry
- It gets the driver request

4.8.8 Euro Radio Interface Manager

The communication protocol is defined by ClearSy. It performs the following tasks:

- It manages the communication with the Euro Radio
- It manages the safe radio connection

- It transmits the radio message

4.8.9 Bitwalker

The Bitwalker is basically the integration layer to the application. It decodes and encodes the bitstream received from the Kernel / operating system into the SRS SUBSET comprehensible message. It will be compliant to all the SUBSETs involved in the communication.

4.8.10 Application Manager

It facilitates the updates of the various modules included in the OBU hardware (Safe Platform).

It shows the application versions available and performs the main functions:

- Installation
- Uninstallation
- Software updates (meaning correction of existing functions)
- Software upgrades (meaning improvements and extensions)

4.8.11 Runtime Environment

The runtime environment (Momentics) delivered by MEN enable an integration with the SIL4 operating system QNX.

4.8.12 Drivers

The system drivers are implemented by MEN. The different Phases mainly differ in the platform on which the openETCS EVC is delivered, and the environment in which it is integrated. This has also an impact on the software delivered in each phase.

4.9 Phase A

Phase A consists of integrating the openETCS EVC in an existing product of the ClearSy portfolio named Operational Simulator. This simulator is a real time simulation demonstrating how a train can be run on tracks under ERTMS/ETCS supervision. Based on predefined scenarios defining all the events that would occur (ETCS telegrams and messages, signal aspect and point position changes, etc.), the operator runs the train with an interactive train movement control.

Figure 19 Operational Simulator standard execution user interface

The standard Operational Simulator product only contains one online module, regrouping all related ERTMS/ETCS components in one whole application:

- EVC functions
- ETCS DMI
- Simulation software (transmission modules, train dynamics, etc.)

Within the scope of Phase A, the Operational Simulator will be split in three independent applications:

- External ETCS DMI
- openETCS EVC functions
- Simulation software packet including the simulation kernel

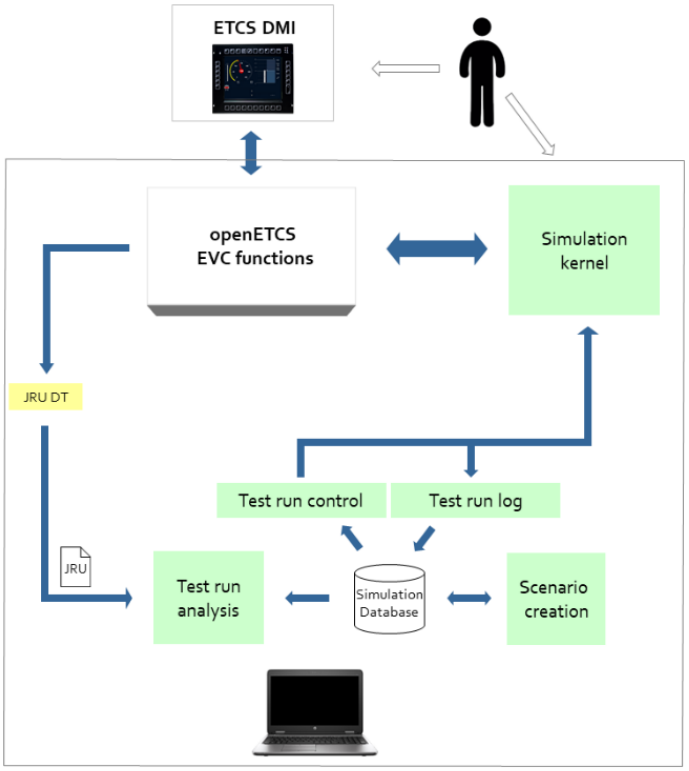


Figure 20 Simulation Phase

4.9.1 Components

- Operational Simulator software:
  - Off line tools
  - On line tools
- openETCS EVC functions (SUBSET-026 version 3.6.0)
- ETCS DMI including the specific application software

4.10 Phase B

Phase B consists of integrating the openETCS EVC in a safe SIL4 platform. An external BTM will be added to the ETCS on-board system architecture as well as a separate JRU. This system has the same architecture as the system deployed on the vehicle during Phase C. The intention however it that it can be tested in the laboratory via a Test Bench, facilitating the integration testing. The Test Bench allows testing an industrial OBU via the usage of interfaces specified in the SUBSET-094 and in the various documents referenced in the SUBSET-094.

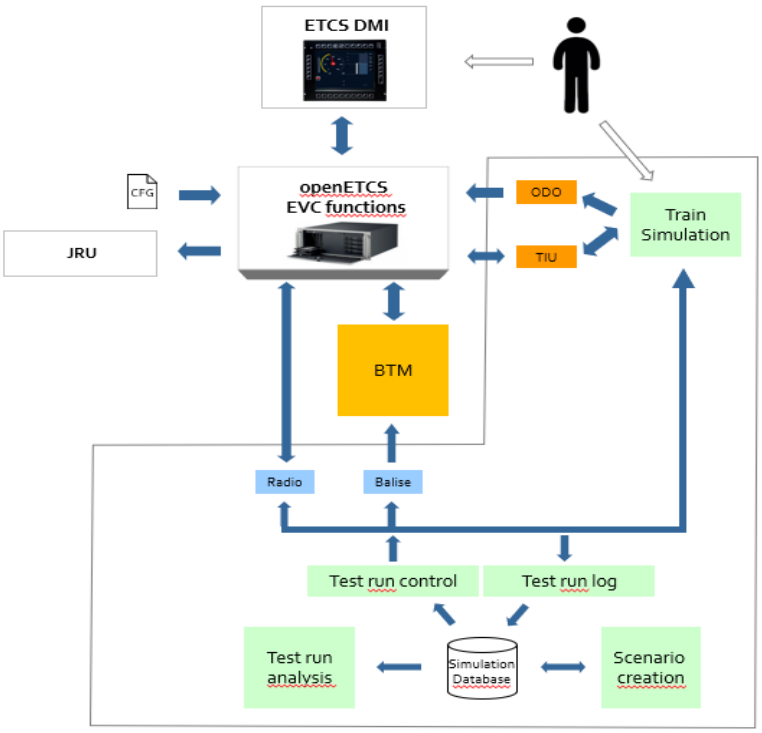


Figure 21 Laboratory Phase

4.10.1 Components

- OBU Test Bench software:
  - Off line tools
  - On line tools, including the BTS software
- OBU Application Manager
- DMI Software
- JRU Software
- BTM Software
- Safe Platform software:
  - QNX
  - openETCS EVC, with kernel according to SUBSET-026 version 3.3.0
  - Odometer (ODO), TIU and other communication modules

4.11 Phase C

Phase C consists of integrating the safe SIL4 platform hosting the openETCS EVC in a real train. The EVC interfaces will no more be connected to the Test Bench but to the real train components.

4.11.1 Components

Since the safe SIL4 platform is integrated in a real train, no specific test environment modules will be used to perform the test. The only exception is the RBC Message Simulator in charge of sending predefined radio messages to the openETCS EVC and receiving radio messages from the openETCS EVC.

The following ETCS on-board system and simulation software modules are involved:

- RBC Message Simulator
- DMI Software
- JRU Software
- BTM Software
- openETCS EVC, with kernel according to SUBSET-026 version 3.3.0

4.12 Phase D

Phase D consists of developing a certain degree of test automation in order to test the openETCS EVC deployed on the safe SIL4 platform. Apart from the EVC the ETCS on-board system architecture involves the external BTM, the separate JRU, the external DMI and the external odometry sub-systems. This system will be designed so that it can be tested using the Test Bench. The Test Bench is compliant with the Reference Architecture defined in SUBSET-094. It allows testing an industrial OBU via the usage of interfaces specified in the SUBSET-094 and in the various documents referenced in the SUBSET-094.

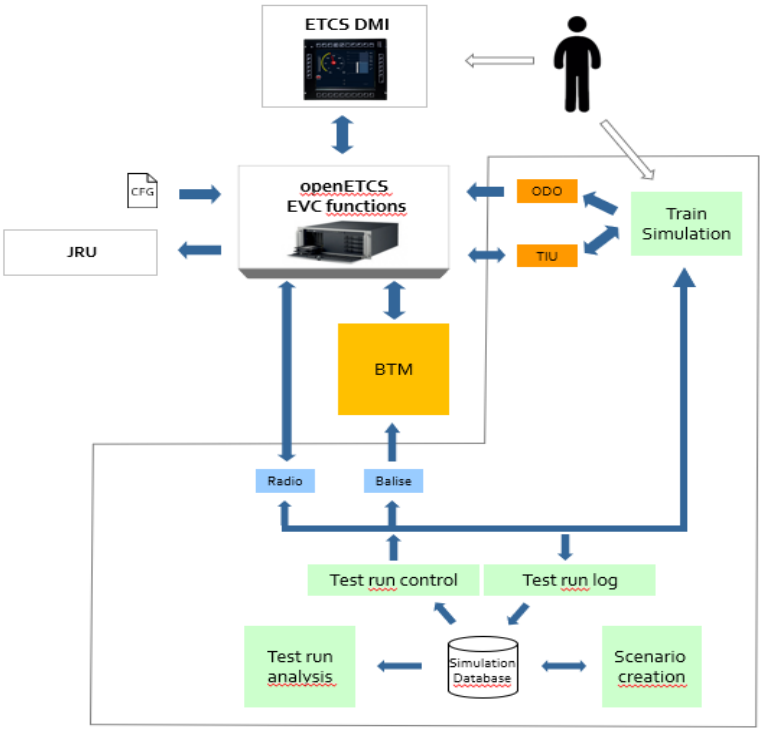


Figure 22 Laboratory Phase

4.12.1 Components

- Test Bench software:
  - Off line tools
  - On line tools, including the BTS software
- Test Bench Application Manager
- DMI Software
- JRU Software
- BTM Software
- Safe Platform integrating the following components:
  - Software:
    - openETCS EVC, with kernel according to SUBSET-026 version 3.3.0
    - QNX
  - Odometer
  - TIU for wired connections
  - Ethernet based communication modules (DMI, JRU, simulator)
  - Further communication modules (BTM, internal communication between components)

4.13 Summary

The following table summarizes the different software modules used within the scope of each Phase:

Project Phase		Phase A	Phase B	Phase C	Phase D
openETCS EVC platform		Laptop	SIL4 platform	SIL4 platform	SIL4 platform
Test environment		Operational Simulator	OBU Test Bench	Real track	OBU Test Bench
Modules	**Database Manager**	✓	✓	✓	✓
	Track Editor	✓	×	×	×
	Scenario Editor	✓	×	×	×
	Scenario Controller	✓	×	×	×
	openETCS Kernel	✓	✓	✓	✓
	Operational Simulator Software	✓	×	×	×
	RBC Message Simulator	×	✓	✓	✓
	OBU application manager	×	✓	✓	✓
	OBU ETCS/ERTMS Software	×	✓	✓	✓
	DMI Software	✓	✓	✓	✓
	OBU Software	×	✓	✓	✓
	QNX	×	✓	✓	✓
	JRU Software	×	✓	✓	✓
	BTM Software	×	✓	✓	✓
	Scenario Analyzer	✓	×	×	×

Figure 23 Software modules delivered in each phase

5 Test Use-Cases

The test scenarios of the Demonstrator are defined in [TEST_002].

6 Conclusions

This chapter will be filled in as soon as the field tests with the demonstrator are finalized.

7 List of Suppliers

This chapter will be filled in as soon as the field tests with the demonstrator are finalized.