# Verification Report for Dynamic ETCS Track Model - Use Case: Amsterdam - Utrecht ETCS L2 reference Line
## Version 0.1

Jan Welte (TU-BS)

Aug 21, 2015

**Abstract**

This document is the verification report for the Dynamic Trackside Model developed in the openETCS project to provide the track information to the ETCS OBU model.

This is the first Version, which only checked the documentation and the track data on paper. A full verification of the track implementation using SCADE will follow in the next version.

*<track data in SCADE model has to be veriefied>*

# 1 Verification of Dynamic Trackside Model Amsterdam - Utrecht ETCS L2

## 1.1 Identity and Configuration of Verification Object

This report contains the verification results for the dynamic trackside model of the Amsterdam - Utrecht ETCS L2 track build in SCADE Suite.

**Verification object**

Dynamic Test Modell - Use Case Amsterdam - Utrecht
   **Software Component Architecture and Design Specification**
Main Document:
   WP 3 D3.5.3 Trackside "Dynamic ETCS Track Model - Use Case: Amsterdam-Utrecht ETCS L2 Reference Line" – AmsterdamUtrecht.pdf
(`https://github.com/openETCS/modeling/tree/master/model/Scade/System/TracksideDynamicModel/TestTracks/UtrechtAmsterdam_oETCS`)
   Designer: Mairamou Haman Adji (pdf created by Peter Mahlmann)
   Version: August 2015 (*<Missing Version Number>*), github commit fcac4457fe2f937b78333c259826e5b0663f

Additional Documents:

List if Balise Locations – Baliselocations-V2.xlsx
(`https://github.com/openETCS/modeling/blob/master/model/Scade/System/`
`TracksideDynamicModel/TestTracks/UtrechtAmsterdam_Documentation/Baliselocations-V2.`
`xlsx`)
Version: V2, 20 Aug 2015, github commit e9abdd2d781216ee014d15db83cb644ac98ee4a8
Contributor: Mairamou Haman Adji

Table of Cross References Balises Track Amsterdam - Utrecht – XREF_Balises_Amsterdam_Utrecht.xlsx
(`https://github.com/openETCS/modeling/blob/master/model/Scade/System/`
`TracksideDynamicModel/TestTracks/UtrechtAmsterdam_oETCS/XREF_Balises_`
`Amsterdam_Utrecht.xlsx`)
Version: 23 Jul 2015 (<*Missing Version Number*>), github commit af9c5e313cfa208ad573aff598fce35f4bcbadad
Contributor: Mairamou Haman Adji

Table of Cross References RBC Messages Track Amsterdam - Utrecht –
XREF_RBC_Messages_Amsterdam_Utrecht.xlsx
(`https://github.com/openETCS/modeling/blob/master/model/Scade/System/`
`TracksideDynamicModel/TestTracks/UtrechtAmsterdam_oETCS/XREF_RBC_Messages_`
`Amsterdam_Utrecht.xlsx`)
Version: 1 Jul 2015 (<*Missing Version Number*>), github commit 9abb9a6a017059f868edf47de42c771ffd21176a
Contributor: Mairamou Haman Adji, Jakob Gaertner

**Software Component SCADE Implementation**
SCADE Model:

UtrechtAmsterdam_oETCS Track Simulation – UtrechtAmsterdam_oETCS.etp
Version: 25 Aug 2015, github commit daad9c39d2ab4c507923a9ba12f38c2677b56128
Designer: Mairamou Haman Adji, Jakob Gaertner

SCADE Model Documentation:

no generated SCADE Model Documentation available at the moment, basic
Implementation concepts are presented in WP 3 D3.5.3 Trackside
**Software Component Test Specification**
<*no documentation available*>

**Covered Requirement Specifications**

No specific component requirement specification is given for the Dynamic Track
Model In general this component shall provide a track description, which sends
the balise and radio information depending on the train position provides by
a train movement simulation. Balise and radio information shall represent the
existing implementation of the Amsterdam-Utrecht ETCS Level 2 track, which
are given be following documents.

- Track Layout Plan provided (by ProRail – infrastructure management)
  Used for balise and signal locations

- JRU data recording from a trip on the track (by NS – railway operator)
  Used for balise and radio message content

**Related Systems/Components**

The dynamic track model is part of the openETCS test track simulation, which is used to simulate a trip on a track for the openETCS EVC model. It can be used in the SCADE simulation environment as well as independently.

Therefore it is related to the following Components:

- train physical movement
  inputs actual train position as distance traveled

- RBC model
  inputs trigger for radio messages

- BTM and EVC model provides balise and radio information via the input channel

This relations are presented in the Component Specifications for the Dynamic Track.

## 1.2 Software Component Architecture and Design Specification Verification

Verifier: Jan Welte (TU Braunschweig)

The Component Design has been mainly done against the respective specification documents. The SCADE model implementation has only been used to verify the design aspects demonstrated on SCADE model images. The verification has been mainly based on manual checking the documents for consistency and against the reference documents of track layout and JRU recording.

*<in a second step test runs of the SCADE implementation will be perfomed to verify the implementation>*

The Dynamic Track model represents a relative extensive component of the openETCS test environment, D 3.5.3 Trackside contains architecture, design and interface information for the dynamic track. As no superordinate Software Specifications for the Environment model are available, so this document has to be verified in a wider scope. The following subsections present the verification results in accordance with EN 50128 7.3.4.42 for Architecture, Design and Interface Specifications as well as 7.4.4.13 for Component Design Specifications:

### 1.2.1 Internal Consistency

The document is consistent concerning the Architecture, Design and Interfaces for the dynamic track model.

The following minor findings are inconsistent or unclear:

- Figure 5 can not be read and is in no way related to the overall document context

- p16 (d) boilerplate not a unique term

- p16 3.3 first paragraph is a quite subjective without references

- p27 last line - what is a normal user and why can he ignore these packages

- p 41 - what is meant with trackside train automation installations

### 1.2.2 Adequacy fulfilling Software Requirements and Design Specifications

Overall the document addresses consistently and completely the purpose for the track model , but no specific reference are given are referenced against which consistency and completeness could be adequately verified.

**Main result:**

- **Task/ requirements to be solved have no reference and are on different abstraction levels**
  The last part of Ch1 lists 3 tasks to solve, which have no reference to a requirement document. These 3 task a correct, but they present different abstraction levels and need to be defined in relation to the system and component requirements and specifications.

- **some objectives on p14 defined are very broad**
  p14 - "simulation model should allow certification as a verification tool (in context of EN50128)" does not specify a reference to EN 50128 and which object shall be certified and how a verification tool is used.
  p14 - "system should at least cover all functionality of the current state of the art" needs a clear reference which defines the state of the art and in which context.

- **Missing reference for test environment software architecture, design and interface Spec**
  D.3.5.3 Trackside also partially includes the Software Architecture, Design and Interface Specifications for the track and RBC components, but does not provide a description for the overall openETCS test environment. Here a reference including at least an overview about all components, their expected functionality and all interfaces is needed.

- **Track model design sufficient for a dynamic track simulation**
  Overall the dynamic track design is adequate to provide the balise and radio messages for a trip simulation on the test track for which the data is implemented. The track model is able to provide the right information for a location to allow a free movement of the train. Nevertheless, the use of those recorded radio messages from the JRU data limits the radio location mostly to the standard communication for a regular trips as this has been the basis of the JRU data.

- **location accuracy for balises have not been taken into account relocation of balises**
  The reasons for relocating balises are given correctly, but the table and all argumentations do not address the location accuracy (in this case always 2m) of all balises. As some balises have a higher deviation as allowed by their location accuracy all assumptions stay reasonable and the performed relocation is done properly. Nevertheless, the location accuracy has to be address and it has to be stated that it cannot be address during the relocation.

- **Exception for M024 has to be explained in more detail**
  The exception for M024 messages is only presented in a short remark, which does not provide enough information concerning reasons and concrete implementation. As the architecture in figure 11 does not present any direct link between RBC and EVC and the RBC has only a trigger to the Track, how are M024 messages are created and transmitted to the EVC.

- **frequency of sending messages**
  It is not clearly stated whether balise telegrams and radio messages are send only one or over a certain interval, when the sending is enabled by position or trigger.

**Minor findings:**

- **structure of document could changed to better separate test system and simulation specification**
  Chapter 2 already presents a number of details for the actual track specification, this should be the beginning of the actual Component design part, as these are the references for the track data. (maybe place before Chapter 4)
  Chapter 3 contains the overall test environment concept (concrete architecture at the beginning of Ch4). This has to be placed at the Beginning or behind the Use Cases to present the top down concept from the EVC model - use cases - test environment simualtion - track model.
  Section 3.3.2 (as 2.3.2) present the concrete reference data, which has to be in the track model design part.
  Chapter 4.2 already has a good content and basic structure, but it would

help to more precise separate between the functional concept and the concrete implementation. The fundamental daisy-chain description is here a bit to SCADE specific.

Chapter 6 has o be place earlier, as these contains requirements.

- **Use of terms Use Cases, User Stories and Scenarios is inconsistent**
  This is a general problem, which cannot be solved by you. That you should use only the term Use Case to avoid ambiguities.

- **Odometry is a component not a data flow in in simulation concept**
  Figure 11 gives a good overview for the test environment software architecture (which ideally should be referenced to a document giving the full documentation). Here Odometry is stated as the data flow from Train Model to EVC Model. This is principal correct, but the odometer is a component at the train, which in the simulation concept receives the real position from the Train Model and gives it's "measured" position (including accuracies and adding random errors) to the EVC.

- **more detailed information needed how N_PIG is used for balise position**
  the sentence in 4.4.1 concept is to broad. Here is a reference concerning the Subset-026 requirement needed. As details are given in 4.4.2 this information should also be referenced.

- **Version 2.3.0 is not the reason for only 2 balises**
  In 4.4.2 it is stated that groups of 2 balises are used because of Version 2.3.0. This is not the case. The track used this combination as this has been a design decision.

### Additional documents

The additional documents have been verified against the track data specified by the *Track Layout Plan* and the *JRU data recording*.

*List if Balise Locations* has only been checked for the area from BG352 to BGBG 445, as only this line is in focus of this verification and the available *Track Layout Plan* has focused on this area (BG 368 and BG 369 locations are not completely readable on plan). The following deviations have been found in the *List if Balise Locations* from the *Track Layout Plan*.

- BG 380 is on 3841, but the *Track Layout Plan* has it marked as 9840

- BG 385 is on 11084, but the *Track Layout Plan* has it marked as 11086

- BG 431 is on 28779, but the *Track Layout Plan* has it marked as 28780

As these deviations seam to be inconsistencies in the data provided, please indicate clearly on which reference has been used for the *List if Balise Locations*.

The data and all respective calculations in the *Table of Cross References Balises* have been verified and been found correct and reasonable, with exception of the following deviations compared to the *JRU data recording*.

- P137 is transmitted by BG361 not BG360

- P72 is missing in BG372, BG376, BG391, BG406, BG424

- P4 is missing in BG435

- P41 is missing in BG440, BG443

- Linking for BG408 unto BG422 is not announced in respect to LRBG 385

The data in the *Table of Cross References RBC Messagess* has been verified and been found correct and reasonable, with exception of the following deviations compared to the *JRU data recording*.

- LRBG_353_D_00319_2_M032 should have distance 365.7 not 319.2

- LRBG_353_D_00431_0_M024 also includes P58 (Position report Parameters)

- missing M024 with trigger 356000559

- LRBG_360_D_00249_2_M015 is incorrect stated as M024

- LRBG_362_D_00034_7_M024 should have distance 36.8 not 34.7

- LRBG_362_D_00136_4_M024 should have distance 138.0 not 136.4

- missing M024 with trigger 362002008

- LRBG_362_D_00230_7_M003 does not include P3 and P41

- LRBG_362_D_00230_7_M003 should have distance 230.6 not 230.7

- LRBG_362_D_00238_9_M003 does not include P3 and P41, but 2 times P65

- missing M024 with trigger 365000525

- LRBG_369_D_00073_3_M024 should have distance 73.7 not 73.3

- LRBG_341_D_00350_0_M024 should have distance 350.5 not 350.0

- all after LRBG341 are not part of this table

### 1.2.3 Readability and Traceability

All concepts and objects in the track design are uniquely named and naming is consistent with track data available. Name conventions are consistent with ETCS standards or a defined explicitly. Only the use of the terms message, telegram and packet is a some points imprecise.

- packet: is a defined data structure

- telegram: is sent by a single balise

- message: is sent via radio or as a BG message build by telegrams of all BG balises, the contain packets

Only a few abbreviations concerning track data are not defined

- p9 - Lint data need and format is not explained or referenced (reason for neglecting required)

- p11 - km reference should be addressed at this point

- p11 - Lint No data need and format is not explained or referenced (reason for neglecting required)

- P 27 - footnote for RBC is placed wrong

Traceability to software requirements and design specifications for the test environment is not established sufficiently, as these document are not given.
Various references or trace are not given in a sufficient way:

- p6 - Use Cases are related to "test requirements for the Amsterdam-Utrecht ...", this document is not clearly named

- p8 - traces for tasked to requirements are missing

- p9 2.1 - "what the train can see" has to be clearly defined and related to the ETCS system architecture

- p9 2.2 - "set of engineering data, track layout plans, and selected JRU recordings" these documents have to be completely referenced (even if they are not public and available over github)

- p9 2.3.1 - track layout plans have to be clearly referenced

- p11 2.3.3 - JRU data is not referenced (format should be also referenced to Subset-027 for JRU)

- p15 - reference to 3rd iteration has to be given, where is this defined

- TSI CCS is not uniquely referenced and at several points wrongly referenced as CCS TSI

- Figure 7 should be referenced in the caption

- p18 3.3.1 - script-driven simulation scenarios are named without reference

- p18 18 - 3.3.2 best also giving reference to cross reference file on github

- p27 "... relevant section" clear reference needed

- p28 - "right sequence" has to be defined more clearly (reference to track)

- p28 - "complete enough" has to be defined an referenced to subset-026 if possible

- p35 - "using available distance information" have to be referenced, where these are presented

### 1.2.4 Consideration of hardware and software constraints

The software constraince have been taken in to account. Any hardware constraints have not been addressed, as openETCS is not dealing with a specific hardware.

## 1.3 Software Implementation

*<The implementation in SCADE for all Balises and Radio messages have not been verified yet.> <To verifiy the data consistency of the implemented data against the JRU recording the following steps will be performed:*

- *translate the JRU data in SCADE constants*

- *build a SCADE test harness to check to confirm that the same data as specified in the JRU is send by the track and at the expected locations*

*>*
*<Further validation activities will be performed, to demonstrate that the track reacts as required to train movements>*