CLUG Demonstration of Readiness for Rail – CLUG 2.0

D3.4 LOC-OB EXTERNAL INTERFACES ANALYSIS

Due date of deliverable: 31/05/2024
Actual submission date: 28/02/2025

Leader of this Deliverable: Marc SARRAT, SNCF
Reviewed: Y

Document status		
Revision	Date	Description
0.1	03/04/2023	Draft version
0.2	08/04/2024	Review meeting
0.3	30/05/2024	First version
0.4	06/09/2024	Draft second version
0.5	30/09/2024	Review meeting
0.6	10/10/2024	Review meeting second version
0.7	10/10/2024	Version for technical review
0.8	22/11/2024	Technical Review comments (AG)
0.9	04/12/2024	Technical Review comments (VB)
1.0	12/12/2024	Final approved version after quality check
1.1	05/02/2025	Answers on external review
2.0	28/02/2025	Final version submitted to EUSPA

Project funded from the European Union's Horizon 2020 research and innovation programme		
Dissemination Level		
PU	Public	X
SEN	Sensitive, limited under the conditions of the Grant Agreement	
Classified R-UE/EU-R	EU RESTRICTED under the Commission Decision No2015/444	
Classified C-UE/EU-C	EU CONFIDENTIAL under the Commission Decision No2015/444	
Classified S-UE/EU-S	EU SECRET under the Commission Decision No2015/444	

Start date of project: 01/02/2023 Duration: 30 months

CLUG 2.0 has received funding from the European Union's Horizon research and innovation programme under grant agreement **No 101082624**

REPORT AUTHORS

NAME	COMPANY	DETAILS OF CONTRIBUTION
Marc Sarrat / Marielle Petit-Doche	SNCF	V0.1: Draft version: - Plan - § Introduction - § System Definition - § Fault tree analysis
Marc Sarrat / Marielle Petit-Doche	SNCF	V0.2: Draft version for review
Marc Sarrat / Marielle Petit-Doche	SNCF	V0.3: First version: remarks from review meeting (13/05/2024)
Marc Sarrat / Marielle Petit-Doche	SNCF	V04: Draft second version
Marc Sarrat / Marielle Petit-Doche	SNCF	V0.5: Second version: remarks from review meeting (30/09/2024)
Marc Sarrat / Marielle Petit-Doche	SNCF	V0.6: Remaining review comments
Marc Sarrat / Marielle Petit-Doche	SNCF	V0.7: Version for technical review
Marc Sarrat / Marielle Petit-Doche	SNCF	V08: Technical review comments (AG)
Marc Sarrat / Marielle Petit-Doche	SNCF	V09: Technical review comments (VB)
Marc Sarrat / Marielle Petit-Doche	SNCF	V1.0: Final version
Marc Sarrat / Marielle Petit-Doche	SNCF	V1.1: Comments from external review

REPORT REVIEWERS

NAME	COMPANY	DETAILS OF CONTRIBUTION
Karin Nebe	SMO	V0.1, V0.4
Claus Thies-Von-Der-Bey	DB Netz	V0.1
Thidarat Panthong	DB Netz	V0.1, V0.4, v0.5, v0.6
Anas Darwich	SNCF	V0.4
Alain Ruaudel	ADS	V0.1
Valentin Barreau	SNCF	V0.8, Technical Review
Adrien Gharios	SNCF	V0.7, Technical Review
Mariya Kayalova	RINA-C	V0.9, Quality check, V2.0 Final quality check
Jose Bertolin	UNIFE	Final check and submission to reviewers and EUSPA

EXECUTIVE SUMMARY

This document is the deliverable “D3.4 – LOC-OB External interfaces Analysis” of the European project “CLUG Demonstration of Readiness for Rail” (hereinafter also referred to as “CLUG 2.0”). This document provides a high level external interfaces safety analysis completed by a safety functional system analysis of the LOC-OB system, identifying how the exchanged between the LOC-OB system and its environment and the internal functional blocks shall be designed to cover the safety requirements already identified in the previous works of the project. This analysis provides too, via an apportionment approach, indications on the selection of safety targets expected on the input measurement means to achieve the TFFR expected on the output functions.

The analysis is performed with the design of fault-trees according to the description of the functional architecture of the LOC-OB. Then qualitative and quantitative analysis are provided to discuss the impact of random failures of the input measurement modules and functional failure of the internal functional blocks on the safety targets expected to the output functions of the system. The analysis is performed for the two solutions of architecture provided by WP4 one with a single chain of computation and one with two different chains of computation.

During the CLUG2.0, project only the functional architecture is defined, no detailed hardware description will be described and analysed. The worst-case approach is followed in this analysis as for the PHA analysis.

In conclusion, this document provides a list of safety requirements to be covered in view of the design of a certifiable product, list of assumptions and a list of open points. The main results of this analysis enhance the importance of safe mechanism for the LOC-OB to receive and provide information. It is confirmed too, that the alternative solution with two diversified chains can allow easily to reach the expected safety target of the LOC-OB. However further work shall analyse in detail, the safe mechanism to compare and combine the results of both chains.

APPLICABLE DOCUMENTS

The following documents define the contractual requirements that all project partners are required to comply with:

- Grant Agreement N°101082624 (which includes Description of Work, Grant Preparation Forms and annexes): This is the contract with the European Commission which defines what has to be done, how and the relevant efforts.
- Consortium Agreement (signed version 13/04/2023): This defines our obligations towards each other.

Each of the above documents was established at the start of the project, and copies were supplied to each partner. Each document could potentially be updated independently of the others during the course of the project following a prescribed process. In the event of any such update, the latest formal issued version shall apply.

In the event of a conflict between this document and any of the contractual documents referenced above, the contractual document(s) shall take precedence.

LIST OF ACRONYMS

ACRONYM	CONCEPTS
CCS	Control, Command and Signalling
CCN	CCS Communication Network
CLUG	Certifiable Localisation Unit with GNSS in the railway environment
CSM	Common Safety Methods
ERA	European Railway Agency
ERTMS	European Rail Traffic Management System
ETCS	European Train Control System
FFFIS	Form Fit Functional Interface Specification
FMEA	Failure Modes and Effects analysis
LOC	Localisation
LWG	Localisation Working Group
OCORA	Open CCS On-board Reference Architecture
ODO	ODOmetry
PHA	Preliminary Hazard Analysis
RAMS	Reliability, Availability, Maintainability and Safety
SCI-*	Standard Communication Interface
SIL	Safety Integrity Level
SRAC	Safety Related Application Condition
TCMS	Train Control Management System
TFFR	Tolerable Functional Failure Rate
THR	Tolerable Hazard Rate

ACRONYM	CONCEPTS
TSI	Technical Specification of Interoperability
TSN	Time-Sensitive Networking
WSoI	Wider System-of-Interest

Contents

1 Introduction 13

1.1 Objectives of the External analysis 13

1.2 Scope of the document 13

1.3 Limits and Hypotheses 14

2 LOC-OB System Definition 15

2.1 OCORA communication network 15

2.2 Interfaces and core components 16

2.3 Detailed Functional Architecture 17

2.4 Alternative architectural solution 18

2.5 Functional blocks 20

3 Safety target Apportionement 25

3.1 PHA results 25

3.2 Apportionment method 26

3.3 Apportionment results on identified architectures 27

3.3.1 Detailed system definition from D4.1 27

3.3.2 Alternative architecture solution analysis 29

4 Fault-Tree analyses 33

4.1 Method 33

4.2 Fault-Tree: based on high level system definition from D2.3 34

4.2.1 Root gate 34

4.2.2 Feared events 36

4.2.3 Common functional blocks 44

4.3 Fault-Tree: based on detailed system definition from D4.1 47

4.3.1 Root gate 47

4.3.2 Feared events 49

4.3.3 Common functional blocks 56
4.3.4 Open issues and assumptions 59
4.4 Alternative architecture solution analysis 61
5 Functional Architecture analyses 63
5.1 Methods 63
5.2 Qualitative analysis 63
5.2.1 High level system definition from D2.3 63
5.2.2 Detailed system definition from D4.1 64
5.2.3 Alternative architecture solution analysis from D4.1 67
5.3 Quantitative analysis 68
5.3.1 High level system definition from D2.3 68
5.3.2 Detailed system definition from D4.1 70
5.3.3 Alternative architecture solution analysis from D4.1 72
6 Synthesis Results 73
6.1 Safety requirements 73
6.2 Assumptions 76
6.3 Open points 77
7 Conclusion 79
8 Reference documents 80

Table of figures

Figure 1: Main External System Constituents and external interfaces (extract from D2.3 [R9]) 17

Figure 2: CLUG LOC-OB functional architecture (adapted from CLUG2.0 D4.1) 18

Figure 3: CLUG LOC-OB double chain functional architecture (adapted from CLUG2.0 D4.1) 19

Figure 4: Fault-Tree for logical decomposition of the detailed architecture Archi_1 (D4.1) 27

Figure 5: Fault-Tree for logical decomposition of the alternative architecture Archi_2 (D4.1) 30

Figure 6: Graphical symbols used in the Fault-Trees 33

Figure 7: Fault-Tree for LOC-OB odometry function according to high-level architecture (D2.3) 35

Figure 8: Fault-Tree for LOC_OB_FE_01 according to high-level architecture (D2.3) 37

Figure 9: Fault-Tree for LOC_OB_FE_02 according to high-level architecture (D2.3) 38

Figure 10: Fault-Tree for LOC_OB_FE_03 according to high-level architecture (D2.3) 39

Figure 11: Fault-Tree for LOC_OB_FE_05 according to high-level architecture (D2.3) 40

Figure 12: Fault-Tree for LOC_OB_FE_06 according to high-level architecture (D2.3) 41

Figure 13: Fault-Tree for LOC_OB_FE_07 according to high-level architecture (D2.3) 42

Figure 14: Fault-Tree for LOC_OB_FE_17 according to high-level architecture (D2.3) 43

Figure 15: Fault-Tree for LOC_OB_SF_302 Compute integrity according to high-level architecture (D2.3) 44

Figure 16: Fault-Tree for LOC_OB_SF_301 Compute Navigation according to high-level architecture (D2.3) 45

Figure 17: Fault-Tree for LOC_OB_SF_303 FDE according to high-level architecture (D2.3) 46

Figure 18: Fault-Tree for LOC-OB odometry function according to detailed architecture (D4.1) 48

Figure 19: Fault-Tree for LOC_OB_FE_01 according to detailed architecture (D4.1) 49

Figure 20: Fault-Tree for LOC_OB_FE_02 according to detailed architecture (D4.1) 50

Figure 21: Fault-Tree for LOC_OB_FE_03 according to detailed architecture (D4.1) 51

Figure 22: Fault-Tree for LOC_OB_FE_05 according to detailed architecture (D4.1) 52

Figure 23: Fault-Tree for LOC_OB_FE_06 according to detailed architecture (D4.1) 53

Figure 24: Fault-Tree for LOC_OB_FE_07 according to detailed architecture (D4.1) 54

Figure 25: Fault-Tree for LOC_OB_FE_17 according to detailed architecture (D4.1) 55

Figure 26: Fault-Tree for Compute integrity according to detailed architecture (D4.1) 56

Figure 27: Fault-Tree for Compute Navigation according to detailed architecture (D4.1) 57

Figure 28: Fault-Tree for FDE according to detailed architecture (D4.1) 58

Figure 29: Fault-Tree for the dual chain according to alternative architecture (D4.1) 62

List of tables

Table 1: Safety requirements related to SCI 16

Table 2: LOC-OB functions identification 24

Table 3: Risk Assessment of LOC-OB (extract from D3.2 [R12]) 25

Table 4: Safety requirements of LOC-OB (extract from D3.2 [R12]) 26

Table 5: Safety target apportionment on detailed architecture (D4.1) 28

Table 6: List of safety requirements identified for apportionment on Archi_1 29

Table 7: Safety target apportionment on detailed architecture (D4.1) 31

Table 8: List of safety requirements identified for apportionment on Archi_2 32

Table 9: Open issues related to the Fault tree analysis 60

Table 10: Minimal cut of order 1 from High Level analysis 64

Table 11: Minimal cut of order 1 according to detailed architecture (D4.1) 65

Table 12: Minimal cut of order 2 according to detailed architecture (D4.1) 66

Table 13: Minimal cut of order 1 according to alternative architecture (D4.1) 67

Table 14: Minimal cut of order 2 according to alternative architecture (D4.1) 67

Table 15: Probability of failure or failure rate of the events for High Level analysis 69

Table 16: Probability of failure or failure rate of the events for detailed architecture (D4.1) 72

Table 17: Probability of failure or failure rate of the events for alternative architecture (D4.1) 72

Table 18: List of safety requirements identified for apportionment on Archi_1 73

Table 19: List of safety requirements identified for apportionment on Archi_2 74

Table 20: List of safety requirements identified. 75

Table 21: Assumptions..... 76

Table 22: Open points 78

1 INTRODUCTION

1.1 Objectives of the External analysis

The objective of this task is to complete the PHA defined in T3.2 by a detailed safety analysis on the external interfaces of the system. The analysis focuses on how the data are exchanged with the external actors and on the safety impacts of the inputs on the data computation up to the outputs of the LOC-OB system.

According to the architectures provided by D2.3 [R9], this task identifies a list of safety requirements on the system and particularly on the interfaces linked to a modular onboard architecture as OCORA (Data exchanged, Standard Communication Interface, etc). However, mechanisms related to the external interfaces (input and output functions) have not been detailed in D4.1 [R14] and the documents provided in WP4. Thus the external safety analysis done here will be limited to the information provided by D2.3 [R9] as done in section 2.1.

To go further in the system analysis of the LOC-OB, a safety system function analysis is provided in the document: its objective is to identify the impact of functional blocks on the failure of the LOC-OB system to provide safe outputs. This analysis is performed for the two architecture solutions proposed in D4.1 [R14].

1.2 Scope of the document

The document focuses first on an analysis based on the high-level architecture described in D2.3 [R9] then on the architecture detailed in D4.1 [R14]. As the analysis is related to the external interfaces, the system description stays at a high level of functional description. An alternative architecture defined in D4.1 [R14] is also discussed.

The apportionment and the fault trees defined in this document are also a base of the analysis of D3.3 [R13].

After a remind of the LOC-OB system definition in section 2, section 3 defines the apportionment of the safety target identified in T3.2 on the functional block of the system. This is a top-down approach that starts from the feared events allocated to the system and defines safety target on the identified functional blocks.

Then, section 4 gives the involvement of the input interfaces to provide the outputs of the LOC-OB system. This is done using Fault-Tree analysis approach. Then a detailed analysis of the failure mode of the external interfaces and the impact of random failure on the input measurements is given in section 5, and the results of these analyses are given in section 6.

1.3 Limits and Hypotheses

During the CLUG2.0, project only the functional architecture is defined, no detailed hardware description will be described and analyzed.

The worst-case approach will be followed in this analysis as for the PHA analysis.

The analyses described in this deliverable are based on a draft and stable version of D4.1 [R14], as the official final version of this document is not yet available. However, the elements form D4.1 used in this document (functions identification and the two functional architectures proposed) are stable and any change on D4.1 will not significantly impact this analysis.

2 LOC-OB SYSTEM DEFINITION

The LOC-OB system definition considered in this analysis is succinctly described in the RAMS Plan [R11], and in more detailed in the specification documents [R8] and [R9].

The main characteristics of the LOC-OB system under analysis are:

- on-board multi-sensor safe localisation system consisting of a navigation core combining GNSS, Inertial Measurement Unit (IMU) and digital map information among others,
- continuous on-board localisation providing position, speed, movement direction acceleration and track selectivity,
- localisation system that is operational and interoperable across the entire European rail network,
- localisation system that is compatible with European Railway Traffic Management System (ERTMS) TSI current status and future evolutions.

2.1 OCORA communication network

As an expected component of the OCORA architecture [R5], the LOC-OB system shall follow the principles of interfaces. and communication between components defined in OCORA [R6], which is the more mature work to define an on-board communication network. This will be updated in the future description of an ETCS CCS-OnBoard in subset 147.

The CCS Communication Network (CCN) allows direct communication between all CCS On-Board components connected to it and eventually with external systems, such as the TCMS. The CCN is the most central part of the OCORA architecture and is the basis for achieving modularity that results in “plug & play” -like exchangeability of all identified building blocks. The CCN is a TSN Ethernet based network with the use of SDTv2/v4 as safety layer.

The Standard Communication Interface provides data from external systems of the LOC-OB and vice-versa. Information can be pulled, pushed, or continuously received through an interface by a defined format over a standardized communication system, i.e., network bus.

These interfaces are specified within OCORA to ensure the modularity and interdependencies of the different subsystems. An interface is understood in this context as a message format defining the information that is exchanged and the procedure, protocol, or physical transmitter. The data exchange between modular services over a centralized communication network enables efficient distribution and usage of information and the specialization of the services, such as localisation.

Within this project, all communication between LOC-OB and the WSoI shall be standardised in the form of SCIs (see [R9]).

The communication interfaces can either be an input, output or bidirectional link. SCIs can be only accessed via the subjacent CCS Communication Network (CCN).

The following table gives a set of safety requirements related to the exchange of the information in safety between LOC-OB and the neighbours systems.

Id	RAMS requirements	Origin	Comments
RA-RAMS-FTA-01	LOC-OB, user equipment and provider equipment shall use data exchange mechanisms in accordance with the safety, security and interoperability requirements.	D2.4	See SpecSysReq[035] in [R10]
RA-RAMS-FTA-02	LOC-OB shall provide its dataset in compliance with the future TSI through the SCI - Vehicle Locator (SCI-VL) interface with a 1.0e-9 ≤ TFFR <1.0e-8.	D2.4	See SpecSysReq[036] in [R10]
RA-RAMS-FTA-03	LOC-OB shall receive safely data sent from the neighbors' on-board system in compliance with the future TSI through the dedicated SCI interfaces with a 1.0e-9 ≤ TFFR <1.0e-8.	D2.4	See SpecSysReq[037], SpecSysReq[038], SpecSysReq[039], SpecSysReq[040], SpecSysReq[041], SpecSysReq[042], SpecSysReq[043], SpecSysReq[044], SpecSysReq[045], SpecSysReq[046] in [R10]

Table 1: Safety requirements related to SCI

2.2 Interfaces and core components

This document focuses on the functional blocks and main interfaces of the LOC-OB system, a synthesis of these interfaces and the high-level architecture of LOC-OB is given in here, for more details see D2.3 deliverable [R9].

An overview of the interfaces is given in Figure 1 showing the input, output and bidirectional interfaces.

Besides to link the input elements to the output, components of the LOC-OB architecture are identified in this Figure 1. They are assigned functional responsibilities, and it is defined, which functions provide what kind of information over which interface.

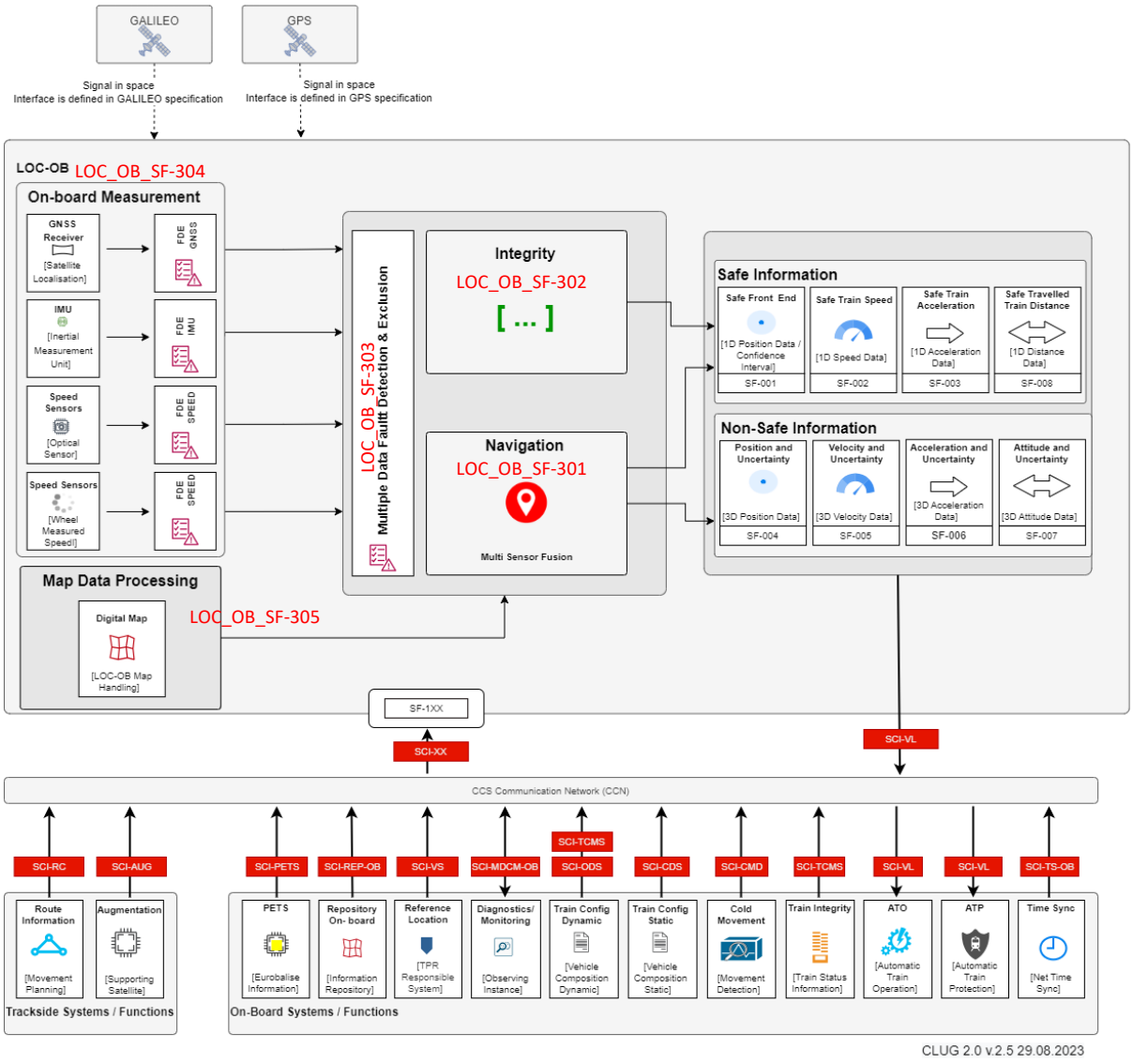

Figure 1: Main External System Constituents and external interfaces (extract from D2.3 [R9])

A short description of the functional blocks is given in §2.5.

2.3 Detailed Functional Architecture

A detailed functional architecture Archi_1 is given in D4.1 [R14].

The Figure 2 gives the main functional modules of the LOC-OB and the data exchanged between them that will be used in the sequel of the safety analysis. It is derived from the description of D4.1 [R14].

The same level of details is kept for the core functions (in green on Figure 2) then for the high level architecture on Figure 1. The selection of measurement types is described in detailed functional architecture.

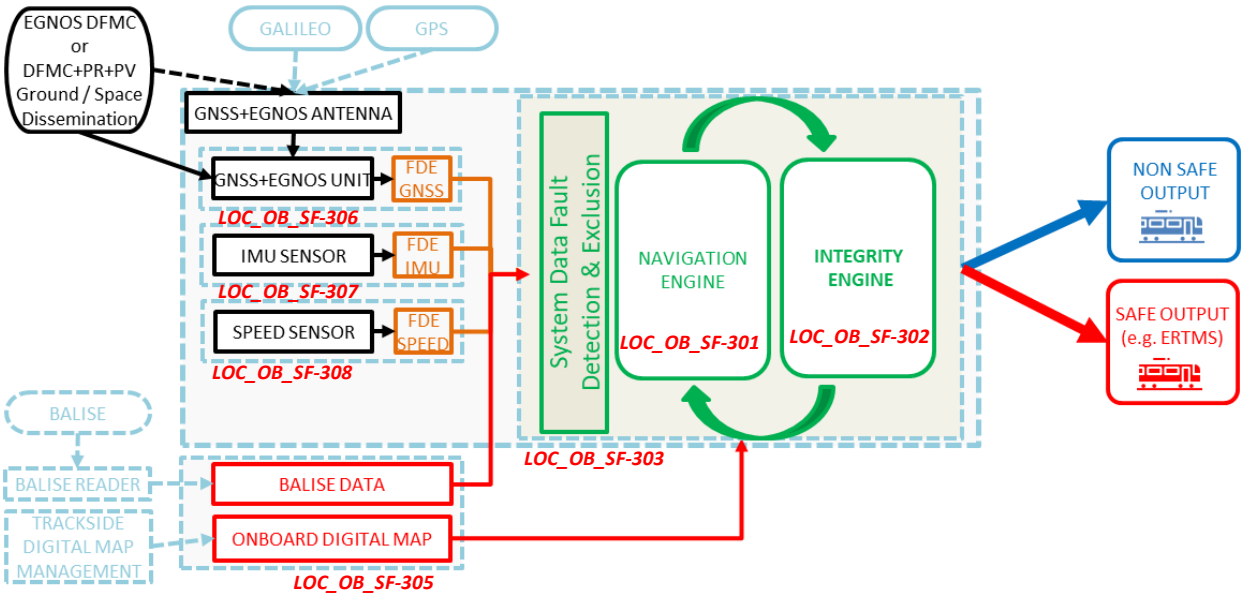

Figure 2: CLUG LOC-OB functional architecture (adapted from CLUG2.0 D4.1)

This chain of computation is composed of three types of measurements:

- GNSS + EGNOS measurements
- IMU measurements
- Speed sensor measurements

It uses as inputs at least:

- Data from balise (received from a balise reader)
- On-board Digital Map

A short description of the functional blocks is given in §2.5.

2.4 Alternative architectural solution

To ensure the target of safety and availability of the LOC-OB system, it is proposed in D4.1 to add to the previous description a second independent chain of computation and a combiner of the results of the both chains as described in Figure 3. It is identified in this document as Archi_2.

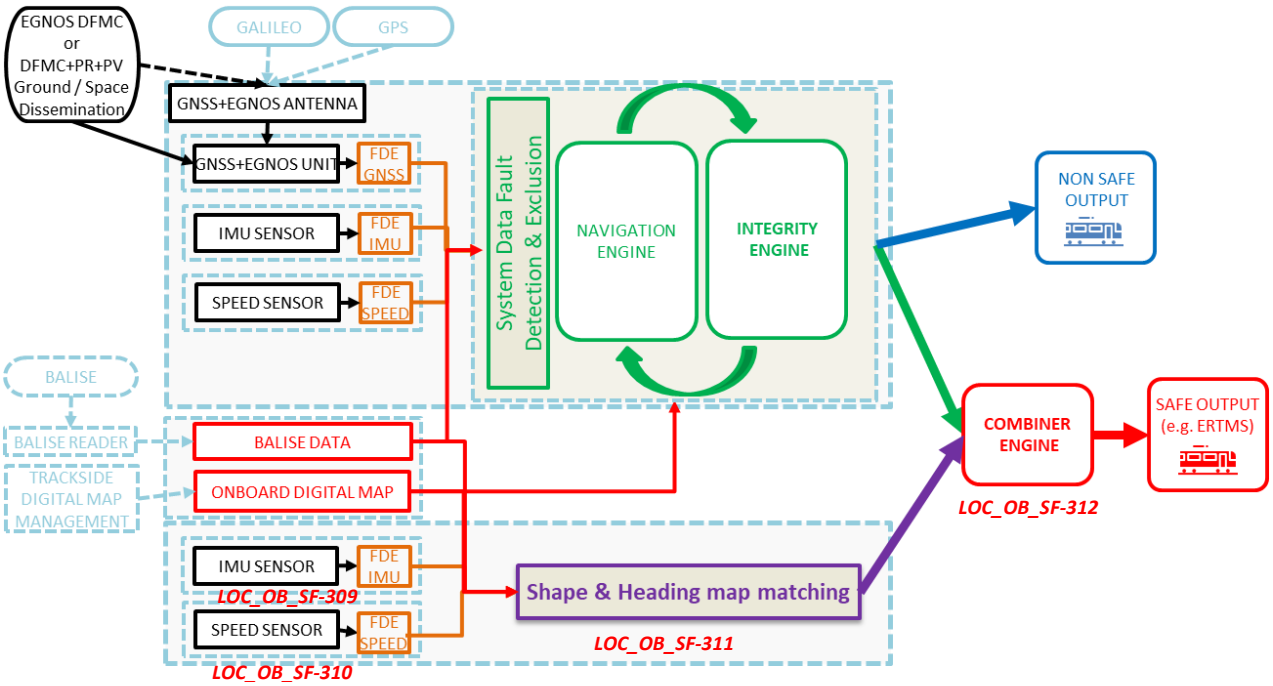

Figure 3: CLUG LOC-OB double chain functional architecture (adapted from CLUG2.0 D4.1)

The first chain is the one described in §2.3.

The second chain is based on a shape map matching algorithm and is composed of two types of measurements:

- IMU measurements
- Speed sensor measurements

Both chains use the same inputs:

- Data from balise (received from a balise reader for the initialization purpose or in area with high performance necessity)
- On-board Digital Map

Assumption LOC-OB-FTA-Ass-30 For analysis on the chain 2, it is assumed that the second chain need balise information provided in safety by the balise reader.

The safety related outputs of both chains are then merged by a combiner to provide the safety outputs functions.

The second chain and the combiner function will not be detailed during the CLUG 2.0 project.

A short description of the functional blocks is given in §2.5.

2.5 Functional blocks

The following table gives a list of the functions of the LOC_OB system. The output functions (identified as LOC_OB_SF-0xx), the input functions (identified as LOC_OB_SF-1xx) and the bidirectional functions (identified as LOC-OB_SF-2xx) are those defined in [R9] and which appear on Figure 1 or Figure 2.

To facilitate the analysis, some identifiers are given to some internal functional blocks as LOC-OB_SF-3xx. These identifiers are added in the Figure 1 and Figure 2. This description is also an input for the FMEA analysis of D3.3 [R13].

Functions Id	Functions	Short Description	Origin/ Assumptions
LOC-OB_SF-001	Provide Safe Train Front End 1D Position Dataset	1D position dataset with safe confidence interval delivered with respect to the last reference location.	Output function, defined in D2.3 [R9]
LOC-OB_SF-002	1D speed with safe confidence interval	1D speed with safe confidence interval.	Output function, defined in D2.3 [R9]
LOC-OB_SF-003	1D acceleration with safe confidence interval.	1D acceleration with safe confidence interval.	Output function, defined in D2.3 [R9]
LOC-OB_SF-004	Provide 3D Position and Uncertainty	3D position and uncertainty.	Output function, defined in D2.3 [R9]
LOC-OB_SF-005	Provide 3D Velocity and Uncertainty	3D velocity and uncertainty.	Output function, defined in D2.3 [R9]
LOC-OB_SF-006	Provide 3D Acceleration and Uncertainty	3D acceleration and uncertainty.	Output function, defined in D2.3 [R9]
LOC-OB_SF-007	Provide 3D Attitude (Rotational Angles) and Uncertainty	3D attitude and uncertainty.	Output function, defined in D2.3 [R9]
LOC-OB_SF-008	Provide Estimated Distance Travelled (since power on)	1D distance delivered in in relation to system initialization.	Output function, defined in D2.3 [R9]
LOC_OB_SF-101	Acquire Map Data	LOC-OB acquires Map Data, e.g., topology and topography information, described through objects defined within an object catalogue.	Input function, defined in D2.3 [R9]
LOC_OB_SF-102	Acquire Route	Route information is a defined sequence of track characteristics that determine the locked or	Input function, defined in D2.3 [R9]

Functions Id	Functions	Short Description	Origin/ Assumptions
		planned path along the railway network represented by a subset of Map Data.	
LOC_OB_SF-103	Acquire Augmentation	The expected augmentation data shall contain information provided by the satellite-based augmentation system (SBAS), e.g., EGNOS, distributed by trackside to establish the independence of the satellite visibility.	Input function, defined in D2.3 [R9]
LOC_OB_SF-104	Acquire Train Integrity	The received data shall include information about train integrity for two use cases: 1) LOC-OB is not installed at the front of the train (cab anywhere); 2) if the train is always connected in regular operation.	Input function, defined in D2.3 [R9]
LOC_OB_SF-105	Acquire static Train Configuration	Expected are the configuration, position, and orientation of the sensors and measurement devices mounted on the train.	Input function, defined in D2.3 [R9]
LOC_OB_SF-106	Acquire dynamic Train Configuration	The following information is expected: train length, status of cabs, rigid definition of the primary moving direction, definition of trains front end.	Input function, defined in D2.3 [R9]
LOC_OB_SF-107	Acquire Eurobalise Telegram	The information is needed to consider passed balises in the multi-sensor fusion, e.g., by linking received balise information with Map Data in the Digital Map. Balise reader will be mandatory in OCORA, thus the identifier and linking information of the balise can be used to compute the localization of the train.	Input function, defined in D2.3 [R9]
LOC_OB_SF-108	Acquire Last Reference Location	The expected information shall contain a reference location on the track network, which LOC-OB uses to provide relative positioning information, e.g., distance.	Input function, defined in D2.3 [R9]

Functions Id	Functions	Short Description	Origin/ Assumptions
		The reference location can be for example a geographical track-bounded point (track edge point) but also a BG.	
LOC_OB_SF-109	Acquire Cold Movement	Expected information is if the train has moved while LOC-OB was not in operation. The LOC-OB will need this information to know if the last location it saved, can still be considered valid or not.	Input function, defined in D2.3 [R9]
LOC_OB_SF-201	Acquire Control Time	Expected is a reference time by a trusted and safe source.	Input function, defined in D2.3 [R9]
LOC_OB_SF-203	Acquire and Provide Control Diagnostics and Maintenance	LOC-OB shall provide data on its own system status and performance and consume diagnostics of the overall CCS-OB.	Input /Output function, defined in D2.3 [R9]
LOC_OB_SF-301	*Compute Navigation (chain 1)*	*The navigation functionality is responsible for the computation of localisation estimated information by processing the standardised data and data from external information sources. A map-assisted multi-sensor fusion algorithm will provide continuously the estimate of the train's position (absolute and relative), speed, acceleration, track selectivity etc. (non-exhaustive list). The resulting information is delivered in a safe and non-safe manner to user functions/systems.*	Internal functional block, defined in D2.3 [R9] and detailed in D4.1 [R14] and D4.6 [R19]
LOC_OB_SF-302	*Compute Integrity (chain 1)*	*The Integrity function is composed of:* • *Data Failure Detection and Exclusion (FDE) functions that are applied to sensor outputs and Navigation functions.* • *Functions that calculate confidence intervals (position, speed, acceleration), confidence status (direction, track selectivity) and data integrity.*	Internal functional block, defined in D2.3 [R9] and detailed in D4.1 [R14] and D4.7 [R20]

Functions Id	Functions	Short Description	Origin/ Assumptions
LOC_OB_SF-303	*System Data FDE (chain 1)*	*System Multiple Data Failure Detection and Exclusion (FDE) function processes and compare all outputs from the measurement functional blocks in order to detect and exclude faulty measurements.*	Internal functional block, defined in D2.3 [R9] and detailed in D4.1 [R14] and D4.6 [R19]
LOC_OB_SF-304	*On-Board Measurement (chain 1)*	*The on-board measurement functionality is responsible for the sensor operation, data acquisition and processing of information on the train movement as input to the sensor fusion algorithm.*	Internal functional block, defined in D2.3 [R9] *This is an overall function decomposed in function LOC_OB_SF-306, LOC_OB_SF-307, LOC_OB_SF-308 in D4.1* ***Remark:*** *All the kind of measurement sensors proposed in D2.3 are not used at the same time. A combination of only 3 sets of sensors or two sets of sensors will be considered in the proposed solutions of D4.1.*
LOC_OB_SF-305	*Map Data Processing*	*Within the LOC-OB, the acquired Map Data needs to be processed according to the use-case of the navigation functionality. For example, Map matching methodologies may be used within the Navigation and Integrity functions.*	Internal functional block, defined in D2.3 [R9] and detailed in D4.1 [R14] and D4.10 [R23] ***Assumption*** **LOC-OB-FTA-Ass-23***: This function covered the specific processing on the map data made in the core of the LOC-OB system. Map Data is provided by the function LOC_OB_SF-101*
LOC_OB_SF-306	*GNSS and SBAS Measurement and FDE (chain 1)*	*This function compiles the GNSS raw data of the GPS and Galileo tracked satellites by the train antenna with the SBAS Aviation Safe augmentation data. This function provides as an output EGNOS-augmented GNSS observables, i.e. corrected clock and satellite positions, error bounds for the corrections and integrity flags. The output data rate is 1Hz. This function associated to a FDE detects and excludes inputs that are not in line with their nominal distributions.*	Internal functional block, defined in D2.3 [R9] and detailed in D4.1 [R14] and D4.2 [R15] ***Assumption*** **LOC-OB-FTA-Ass-24***: In this document the measurement functions cover the sensors, measurement algorithms and Fault detection and exclusion for a given technology.*

Functions Id	Functions	Short Description	Origin/ Assumptions
LOC_OB_SF-307	*IMU Measurement and FDE (chain 1)*	*This function provides the raw measurements from the sensors analysed to detect and to exclude faulty ones.*	Internal functional block, defined in D2.3 [R9] and detailed in D4.1 [R14] and D4.3 [R16] ***Assumption* LOC-OB-FTA-Ass-24***: In this document the measurement functions cover the sensors, measurement algorithms and Fault detection and exclusion for a given technology.*
LOC_OB_SF-308	*Wheel sensors Measurement and FDE (chain 1)*	*This function provides the raw measurements from the sensors analysed to detect and to exclude faulty ones.*	Internal functional block, defined in D2.3 [R9] and detailed in D4.1 [R14] and D4.4 [R17] ***Assumption* LOC-OB-FTA-Ass-24***: In this document the measurement functions cover the sensors, measurement algorithms and Fault detection and exclusion for a given technology.*
LOC_OB_SF-309	*IMU Measurement and FDE (chain 2)*	*This function provides the raw measurements from the sensors analysed to detect and to exclude faulty ones.*	Second chain, internal functional block, defined in D4.1 [R14] and detailed in [R24]
LOC_OB_SF-310	*Wheel sensors Measurement and FDE (chain2)*	*This function provides the raw measurements from the sensors analysed to detect and to exclude faulty ones.*	Second chain, internal functional block, defined in D4.1 [R14] and detailed in [R24]
LOC_OB_SF-311	*Shape and Heading Map matching (chain 2)*	*This function provides the safe 1D localisation information by implementing a shape and heading matching algorithm.*	Second chain, internal functional block, defined in D4.1 [R14] and detailed in [R24]
LOC_OB_SF-312	*Combiner*	*This function combines the outputs of both chains to provide the safe outputs of the LOC-OB i.e. the information provided by:* • *LOC-OB_SF-001* • *LOC-OB_SF-002* • *LOC-OB_SF-003*	Combiner functional block, defined in D4.1 [R14] and D4.7 [R20]

Table 2: LOC-OB functions identification

3 SAFETY TARGET APPORTIONEMENT

The first step of the analysis is to apportion the results of the Preliminary Hazard Analysis [R12] on the functional components of the architecture. After recalling the results of the PHA, this section describes the method followed for the apportionment and the results on the architectures described in section 2.

3.1 PHA results

The Table 3 of the feared events related to the output functions is extracted from the Preliminary Hazard Analysis [R12]. Only the feared events related to the safety appear here, for more details see [R12].

Functions		Feared Events		Design Target TFFR	Comments
LOC-OB_SF-001	Provide Safe Train Front End 1D Position Dataset	**LOC-OB_FE_03**	Fail to provide the correct train orientation	1.0e-9 ≤ TFFR < 1.0e-8	In the PHA, the feared events are linked to erroneous data send by the LOC-OB. It is assumed in PHA, that if outputs data are omitted by the LOC-OB, the users can detect the omission.
		LOC-OB_FE_05	Fail to use the correct reference point id		
		LOC-OB_FE_06	Fail to provide confidence interval which include the real train position (measured distance interval and position qualifier and reference point id and track edge id)		
		LOC-OB_FE_17	Fail to provide the correct track edge id		
LOC-OB_SF-002	1D speed with safe confidence interval	**LOC-OB_FE_01**	Fail to provide upper bound speed higher than the actual train speed	1.0e-9 ≤ TFFR < 1.0e-8	
		LOC-OB_FE_02	Fail to provide the correct train movement direction		
LOC-OB_SF-003	1D acceleration with safe confidence interval.	**LOC-OB_FE_07**	Fail to provide lower bound acceleration lower than the actual train acceleration	1.0e-9 ≤ TFFR < 1.0e-8	

Table 3: Risk Assessment of LOC-OB (extract from D3.2 [R12])

In the sequel, we will consider the feared events related to a 1.0e-9 ≤ TFFR < 1.0e-8 per hour i.e.: LOC-OB_FE_01, LOC-OB_FE_02, LOC-OB_FE_03, LOC-OB_FE_05, LOC-OB_FE_06, LOC-OB_FE_07, LOC-OB_FE_17.

A set of Safety requirements has been deduced from the PHA and are listed in the Table 4: Safety requirements of LOC-OB (extract from D3.2 [R12]):

Id	RAMS requirements	Origin	Comments
RA-RAMS-01	The safety of the LOC-OB shall be ensured and demonstrated according to the Common Safety Methods [ERA CSM] and the [EN 50126] standard.	D2.1	See UR[020] in [R8]
RA-RAMS-02	LOC-OB shall not degrade safety toward odometry as defined in the ETCS BL3 R2.	D2.1	See UR[021] in [R8]
RA-RAMS-03	The true train position shall be always inside the confidence interval.	D2.1	See UR[022] in [R8]
RA-RAMS-04	The true train speed shall be always inside the confidence interval.	D2.1	See UR[023] in [R8]
RA-RAMS-05	The true train acceleration shall be always inside the confidence interval.	D2.1	See UR[024] in [R8]
RA-RAMS-06	Each localisation information shall fulfil safety target requirements in accordance with the consumer's application requirements.	D2.1	See UR[025] in [R8]
RA-RAMS-07	Loc-OB shall respect the standards EN 50121 and EN 50155.	D3.2	
RA-RAMS-08	The track edge ID provided by LOC-OB shall always be the track edge occupied by the train front end real position.	D2.1	See UR[022] in [R8]

Table 4: Safety requirements of LOC-OB (extract from D3.2 [R12])

3.2 Apportionment method

The process of apportionment followed in this document is the one from the standard EN50126 [R2] and EN50129 [R3] as referenced by TSI. More details on the method can be find in the article [R25].

The PHA has allowed to identify the risks related to the LOC-OB, to assess these risks, and then to identify feared events allocated to the output functions (see the results in §3.1). The quantifications related to each function (see §3.1) is the starting point of the apportionment.

Thus, a TFFR is allocated to each output functions of the system.

The following step is to apportion this TFFR to the functional blocks following the identified architecture of the functions. This apportionment follows a logical combination of the functions.

Simplified fault trees are built to perform this apportionment, then rules below are applied to define the objectives of each functional block:

In case of OR-gate: the safety objective is equally apportioned between the functions:

$$TFFR\_system = \sum TFFR\_subsystems$$

In case of AND-gate: the safety objective depends of the time necessary to detect a failure of each subsystem, called Safe down Rate (SDR)and is equally apportioned between functions:

$$TFFR\_system = \prod (TFFR\_subsystems / SDR\_subsystems) * \sum SDR\_subsystems$$

3.3 Apportionment results on identified architectures

3.3.1 Detailed system definition from D4.1

The following figures gives the high level logical decomposition for detailed architecture Archi_1 (see §2.3):

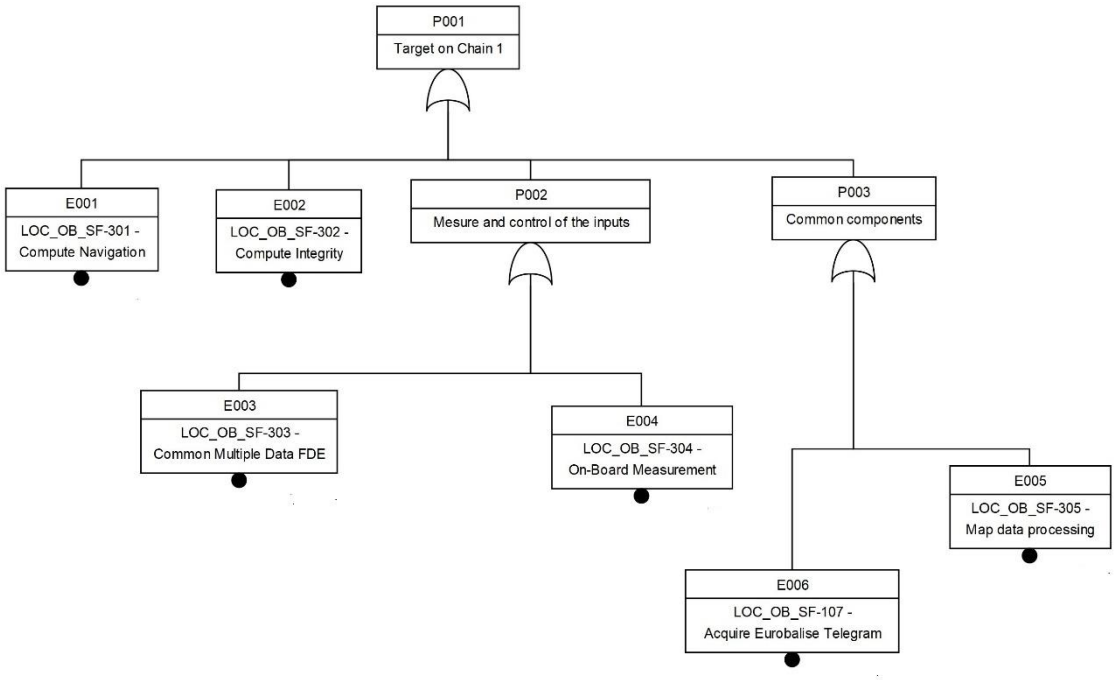

Figure 4: Fault-Tree for logical decomposition of the detailed architecture Archi_1 (D4.1)

In case of one chain architecture the following apportionment is defined according to the ules defined in section 3.2:

Element as identified on Figure 4	Function	TFFR	Comments
P001	Root gate	1.0e-9	Seven feared events are allocated at the top gate, a conservative target for the system is selected.
E001	***LOC_OB_SF-301***	0.25e-9	P001 is equally apportioned between four blocks
E002	***LOC_OB_SF-302***	0.25e-9	P001 is equally apportioned between four blocks
P002		0.25e-9	P001 is equally apportioned between four blocks
E003	***LOC_OB_SF-303***	0.5e-10	P002 target is apportioned mainly to the FDE functional block, which is in charge, in a limited time to detect the errors of the input measurements
E004	***LOC_OB_SF-304***	0.2e-9	P002 target is apportioned mainly to the FDE functional block which is in charge, in a limited time to detect the errors of the input measurements
P003		0.25e-9	P001 is equally apportioned between four blocks
E005	***LOC_OB_SF-305***	0.125e-9	P003 is equally apportioned between three blocks
E006	**LOC_OB_SF-107**	0.125e-9	P003 is equally apportioned between three blocks

Table 5: Safety target apportionment on detailed architecture (D4.1)

Thus, the following safety requirements can be issued:

Id	RAMS requirements	Comments
RA-App-Archi_1-01	For Archi_1, the ***LOC_OB_SF-301*** Compute Navigation function shall be designed in SIL4 with a TFFR ≤ 0.25e-9 per hour on the output	
RA-App-Archi_1-02	For Archi_1, the ***LOC_OB_SF-302*** Compute Integrity function shall be designed in SIL4 with a TFFR ≤ 0.25e-9 per hour on the output	
RA-App-Archi_1-03	For Archi_1, the ***LOC_OB_SF-303*** System Data FDE function shall be designed in SIL4 with a TFFR ≤ 0.5e-10 per hour on the output	
RA-App-Archi_1-04	For Archi_1, the ***LOC_OB_SF-304*** On-Board Measurement function shall be designed in SIL4 with a TFFR ≤ 0.2e-9 per hour on the output	

Id	RAMS requirements	Comments
RA-App-Archi_1-05	For Archi_1, the ***LOC_OB_SF-305*** Map Data Processing function shall be designed in SIL4 with a TFFR ≤ 0.125e-9 per hour on the output	
RA-App-Archi_1-06	For Archi_1, the **LOC_OB_SF-107** Acquire Eurobalise Telegram function shall be designed in SIL4 with a TFFR ≤ 0.125e-9 per hour on the output	This safety requirement is to discuss in regard to the system which provide the Eurobalise Telegram.

Table 6: List of safety requirements identified for apportionment on Archi_1

3.3.2 Alternative architecture solution analysis

The following figure gives the high-level logical decomposition for alternative architecture Archi_2 (see §2.4). Gate P004 is related to the apportionment on the first chain as described in section 3.3.1 Figure 4, except for function LOC_OB_SF-305 which is common to both chains.

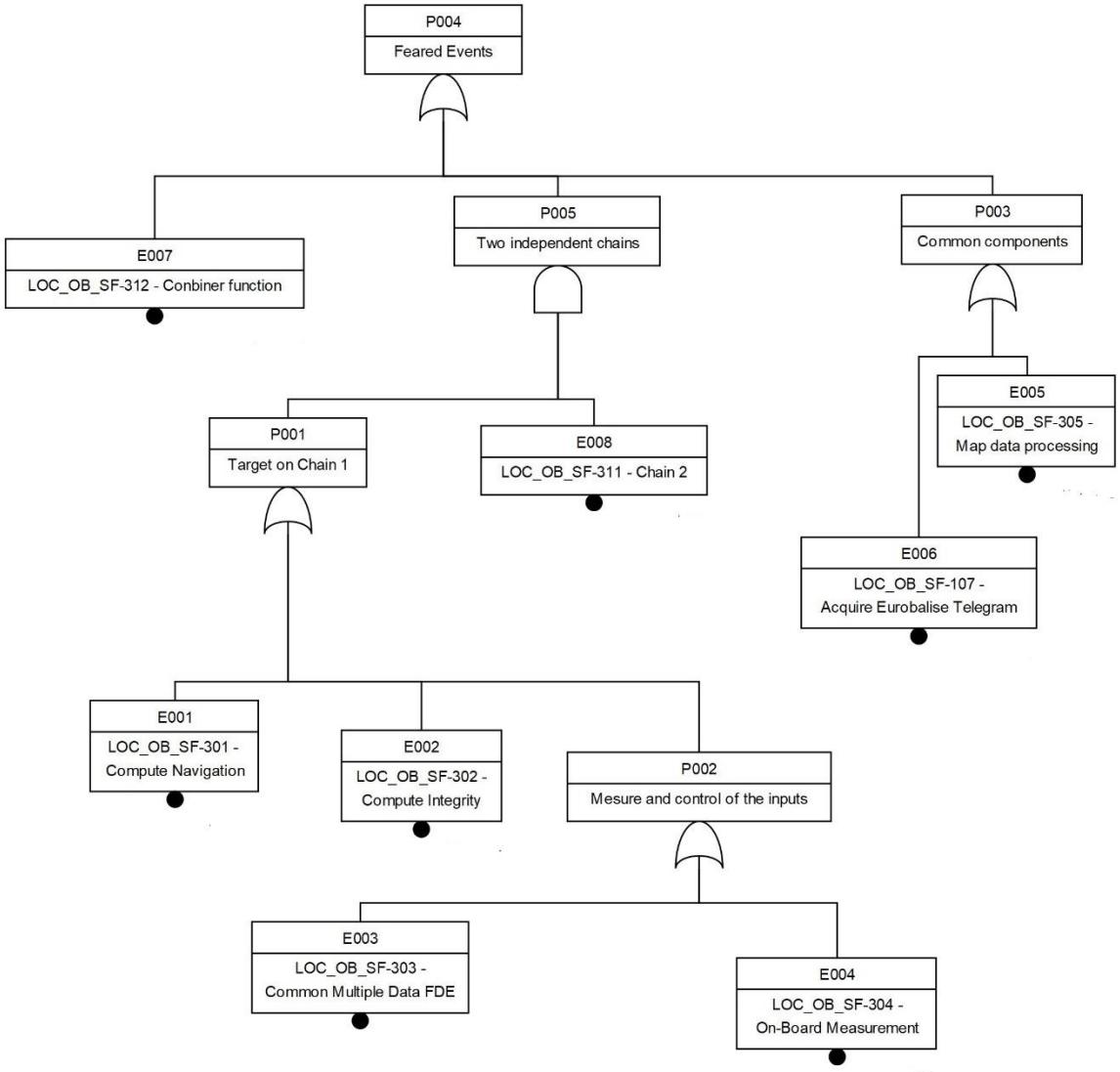

Figure 5: Fault-Tree for logical decomposition of the alternative architecture Archi_2 (D4.1)

Some safety requirements are defined to allow an apportionment of a lower safety target to each chain:

In case of dual chain architecture, the following apportionment is defined, under this assumption:

Assumption LOC-OB-FTA-Ass-31 For apportionment and analysis on the dual chain architecture, and to cover **RA-App-Archi_2-09,** it is assumed that the combiner function can detect and manage the failure of each chain in less the 30 minutes.

Element as identified on Figure 5 and Figure 4	Function	TFFR	Comments
P004	Root gate	1.0e-9	Seven feared events are allocated at the top gate, a conservative target for the system is selected.
E007	***LOC_OB_SF-312***	0.3e-9	P004 is equally apportioned between three blocks
P003		0.3e-9	P004 is equally apportioned between three blocks
E005	***LOC_OB_SF-305***	0.15e-9	P003 is equally apportioned between three blocks
E006	**LOC_OB_SF-107**	0.15e-9	P003 is equally apportioned between three blocks
P005		0.3e-9	P004 is equally apportioned between three blocks
E002	***LOC_OB_SF-311***	1.7e-5	P005 is equally apportioned, taking into account a SDR of 2 , which means 30min to detect and manage the failure of each chain by the combiner see **LOC-OB-FTA-Ass-31**
P001		1.7e-5	P005 is equally apportioned, taking into account a SDR of 2 , which means 30min to detect and manage the failure of each chain by the combiner see **LOC-OB-FTA-Ass-31**
E001	***LOC_OB_SF-301***	5.8e-6	P001 is equally apportioned between three blocks
E002	***LOC_OB_SF-302***	5.8e-6	P001 is equally apportioned between three blocks
P002		5.8e-6	P001 is equally apportioned between three blocks
E008	***LOC_OB_SF-303***	0.8e-6	P002 target is apportioned mainly to the FDE functional block, which is in charge, in a limited time to detect the errors of the input measurements
E009	***LOC_OB_SF-304***	0.5e-5	P002 target is apportioned mainly to the FDE functional block which is in charge, in a limited time to detect the errors of the input measurements

Table 7: Safety target apportionment on detailed architecture (D4.1)

Thus the following safety requirements can be issued:

Id	RAMS requirements	Comments
RA-App-Archi_2-01	For Archi_2, the ***LOC_OB_SF-301*** Compute Navigation function shall be designed in SIL2 with a TFFR ≤ 5.8e-6 per hour on the output	
RA-App-Archi_2-02	For Archi_2, the ***LOC_OB_SF-302*** Compute Integrity function shall be designed in SIL2 with a TFFR ≤ 5.8e-6 per hour on the output	
RA-App-Archi_2-03	For Archi_2, the ***LOC_OB_SF-303*** System Data FDE function shall be designed in SIL2 with a TFFR ≤ 0.8e-6 per hour on the output	
RA-App-Archi_2-04	For Archi_2, the ***LOC_OB_SF-304*** On-Board Measurement function shall be designed in SIL2 with a TFFR ≤ 0.5e-5 per hour on the output	
RA-App-Archi_2-05	For Archi_2, the ***LOC_OB_SF-305*** Map Data Processing function shall be designed in SIL4 with a TFFR ≤ 0.15e-9 per hour on the output	
RA-App-Archi_2-06	For Archi_2, the **LOC_OB_SF-107** Acquire Eurobalise Telegram function shall be designed in SIL4 with a TFFR ≤ 0.15e-9 per hour on the output	This safety requirements is to discuss in regard to the system which provides the Eurobalise Telegram.
RA-App-Archi_2-07	For Archi_2, each chain shall provide safe output functions with TFFR ≤ 1.7e-5 per hour.	
RA-App-Archi_2-08	For Archi_2, the ***LOC_OB_SF-312*** Combiner function shall be designed in SIL4 with a TFFR ≤ 0.3e-9 per hour on the output	
RA-App-Archi_2-09	In case of two chains with two independent computation functions and two independent sets of sensors, a combiner function shall implement a mechanism to detect and manage the failure of each chain in a given limited time in view to provide safe outputs.	The time to detect the failure of each chain shall be identified in the specification of the combiner function. A value of half an hour as been used in this analysis which gives conservative value.
RA-App-Archi_2-10	The two chains shall be designed with independent computation functions and independent sets of sensors.	

Table 8: List of safety requirements identified for apportionment on Archi_2

4 FAULT-TREE ANALYSES

4.1 Method

This document follows a Fault-Tree analysis approach: it is a deductive approach largely used in the safety domain, to identify the failure of elements of the system which will lead to the top feared event.

A feared event of the system (see section3.1) is identified as the root (or top event) of the tree. Then logical gates (in this study “OR” or “AND” gates) are selected to identify the combination of causes of this feared event. This building is iterated until elementary events or transfer gates are reached.

Figure 6 gives the description of the graphical symbols used in the document.

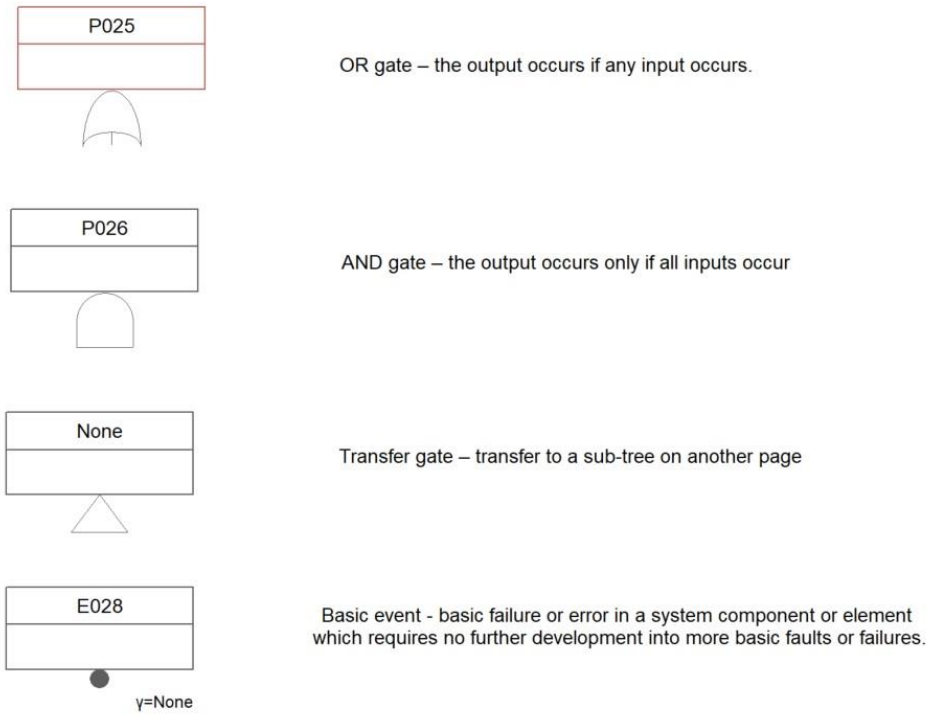

Figure 6: Graphical symbols used in the Fault-Trees

Some qualitative and quantitative analysis can be conducted on the produced Fault-Trees and are discussed in section 5 in view of an external interface analysis.

Thus in the sequel, some labels appear on the gate on the fault-tree:

- On basic event, for which a constant law is associated γ is the probability of failure on demand allocated to the function
- On basic event for which a normal law is associated λ is the failure of the component
- On gate :

- T is the mission time considered for the safety analysis: one year, 24h by day gives 8760 hours
- L is the failure rate on this gate at T hours
- L_avg is the mean failure rate during T

This Fault-Trees analysis will be completed by an FMEA in the document [R13].

4.2 Fault-Tree: based on high level system definition from D2.3

This section gives a set of trees deriving the feared events of section 3.1 according to the high-level architecture defined in D2.3 [R9] (see figure Figure 1). Qualitative and quantitative analyses will be given in sections 5.2.1 and 5.3.1.

4.2.1 Root gate

This first fault-tree contents the root gate for the LOC-OB system: it synthesizes the safety feared events which leads to a failure of the odometric function.

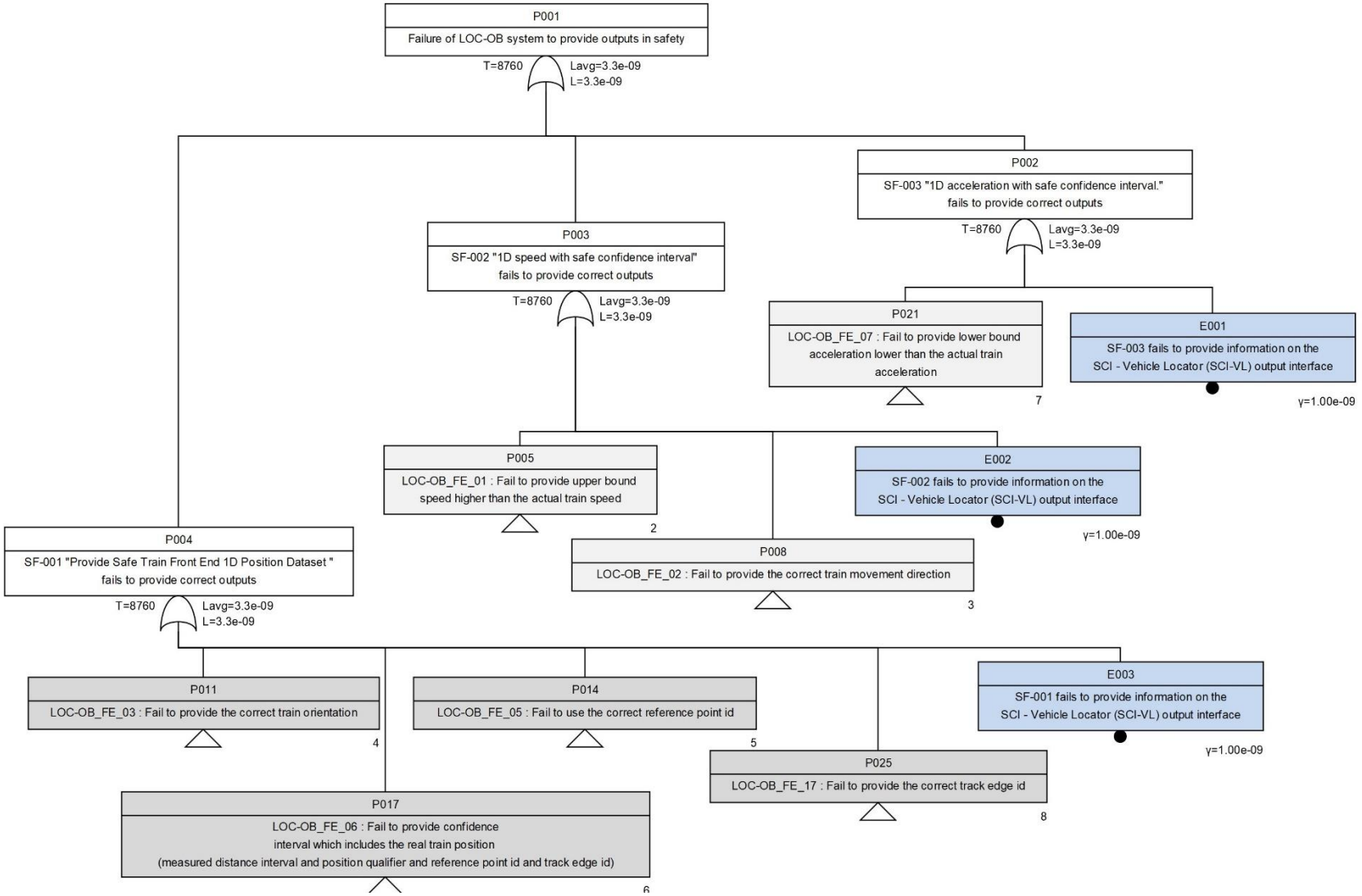

Figure 7: Fault-Tree for LOC-OB odometry function according to high-level architecture (D2.3)

4.2.2 Feared events

This section gives the list of the Fault-Trees derived from the feared events listed in §3.1, according to the high-level architecture from D2.3.

Some trees related to the main functions (Navigation, integrity and on-board measurement) are defined in §4.2.3.

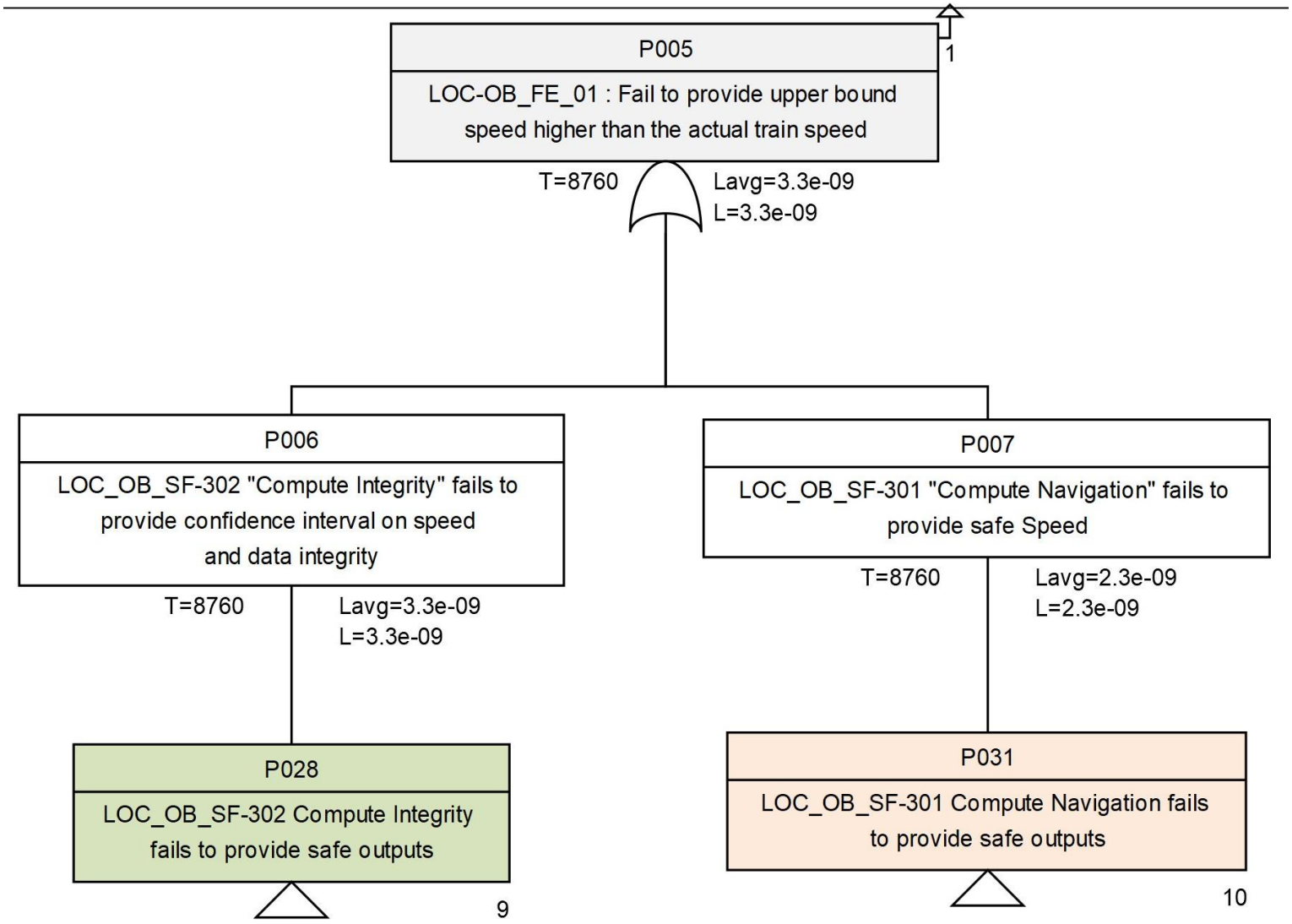

Figure 8: Fault-Tree for LOC_OB_FE_01 according to high-level architecture (D2.3)

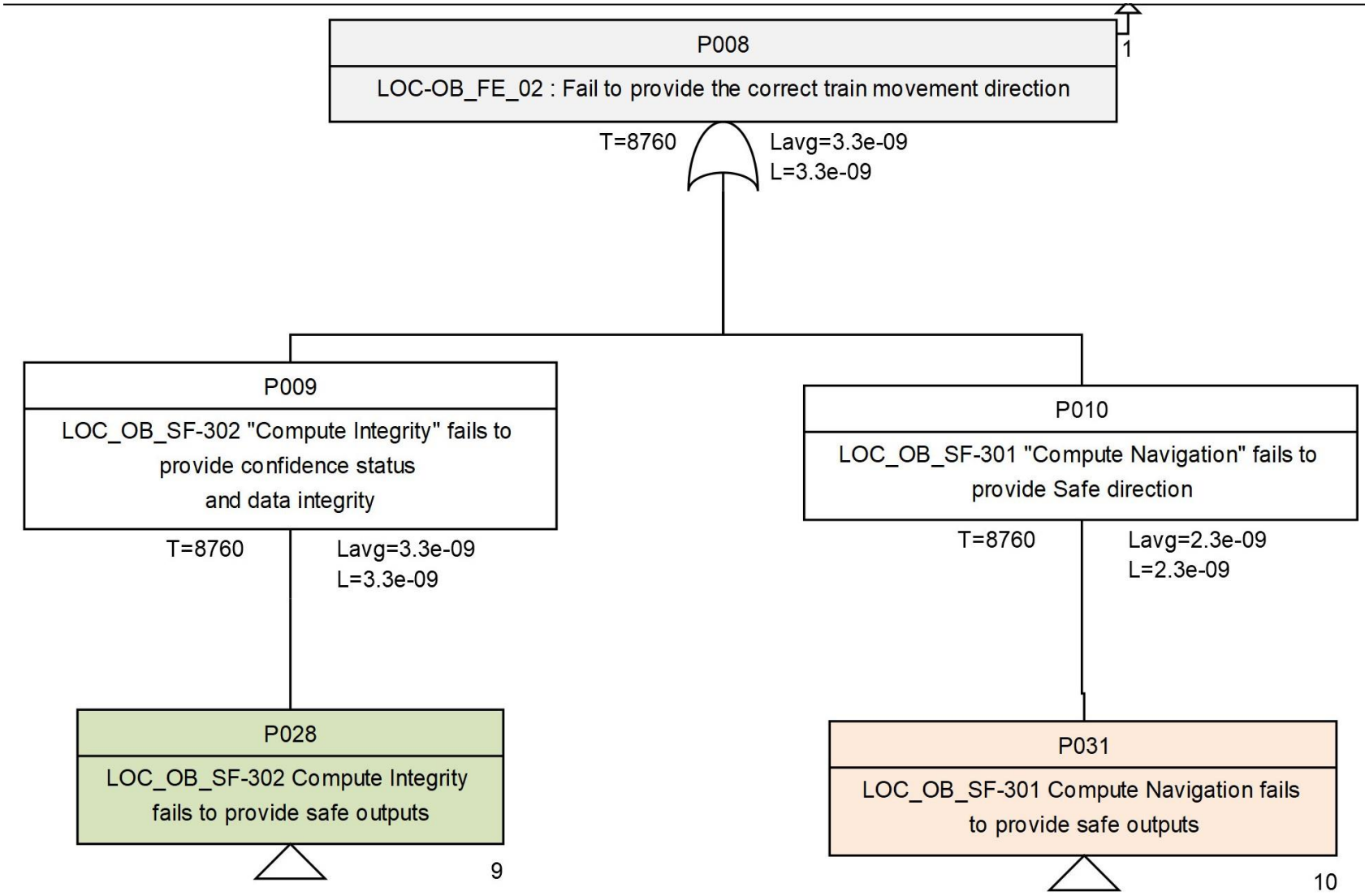

Figure 9: Fault-Tree for LOC_OB_FE_02 according to high-level architecture (D2.3)

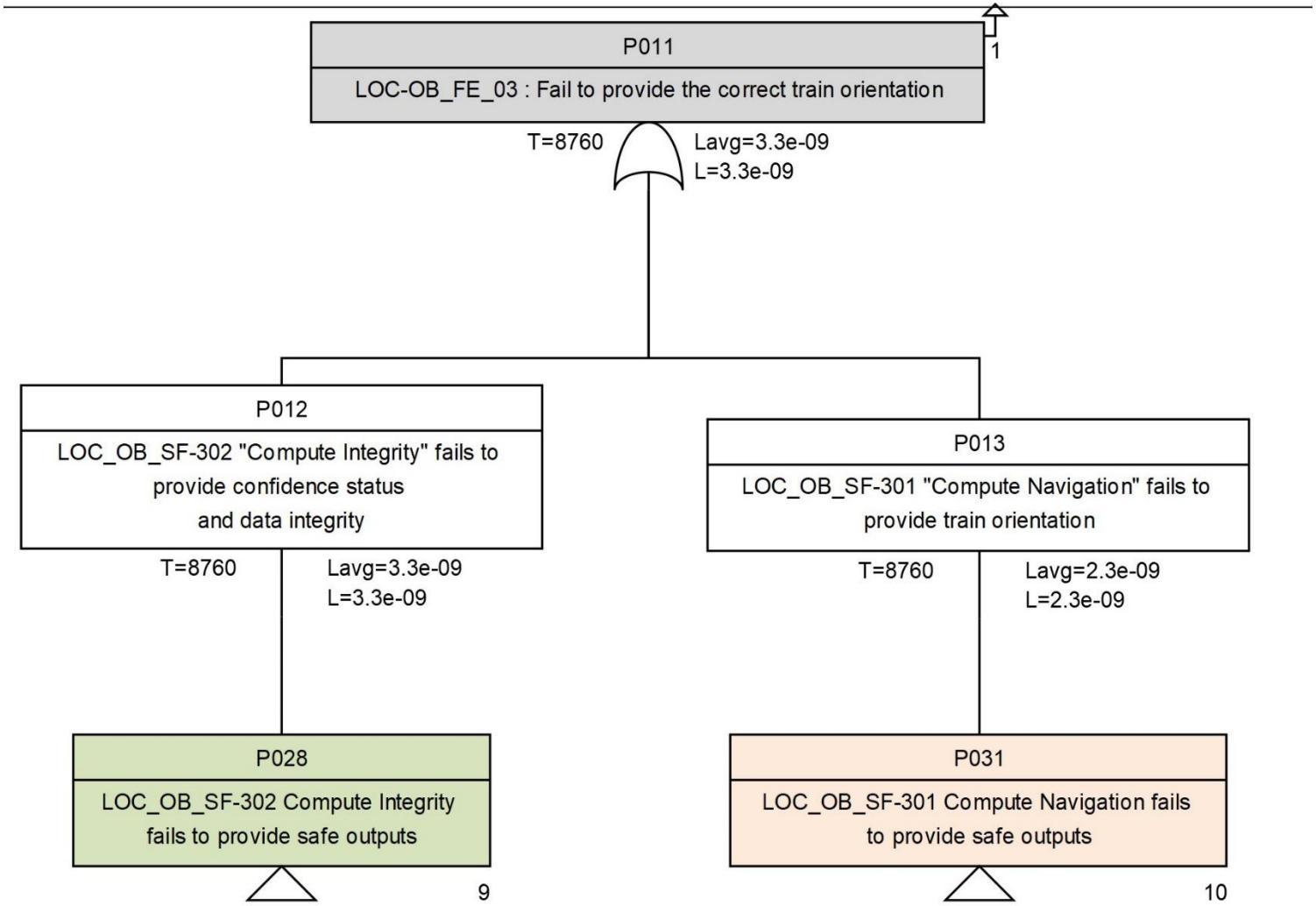

Figure 10: Fault-Tree for LOC_OB_FE_03 according to high-level architecture (D2.3)

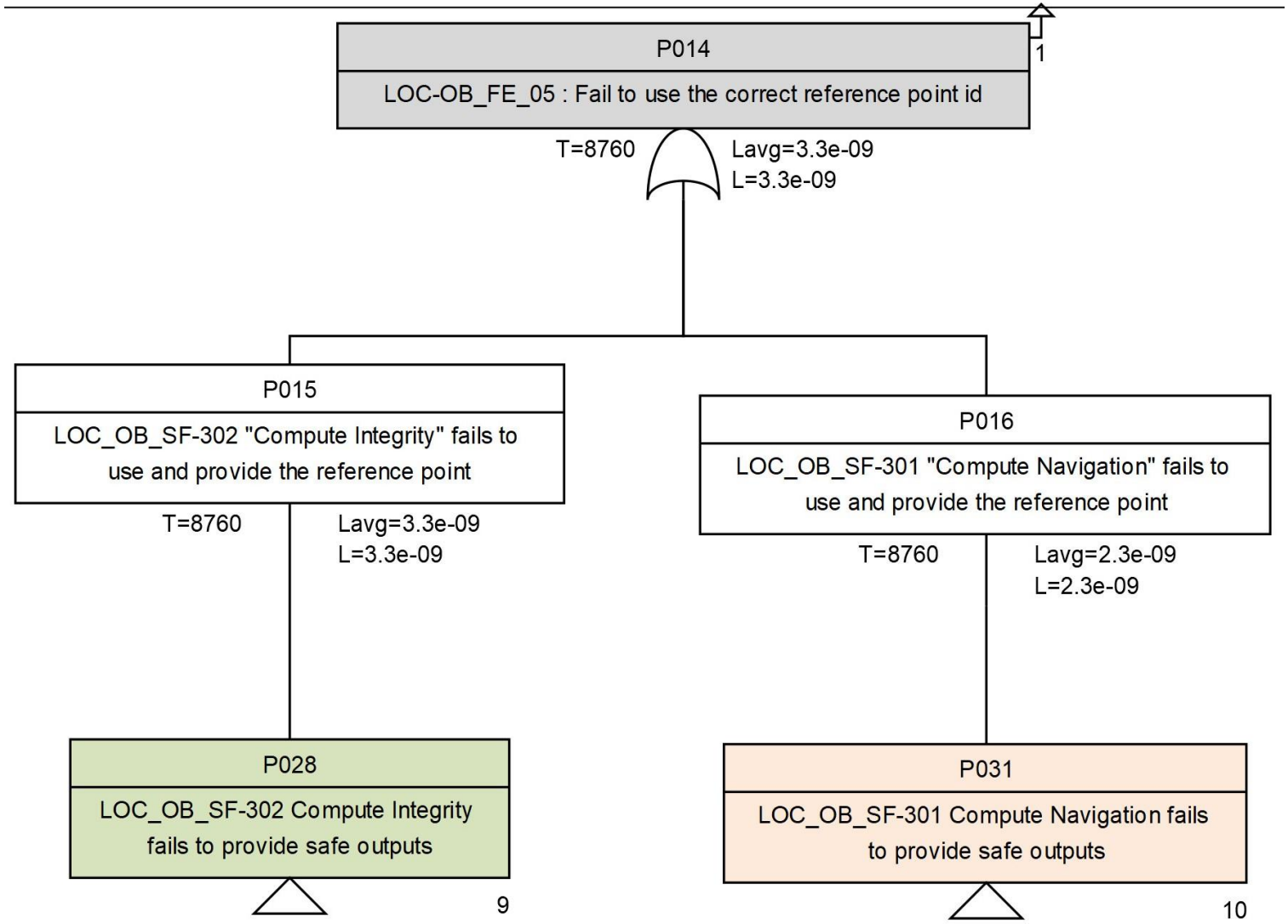

Figure 11: Fault-Tree for LOC_OB_FE_05 according to high-level architecture (D2.3)

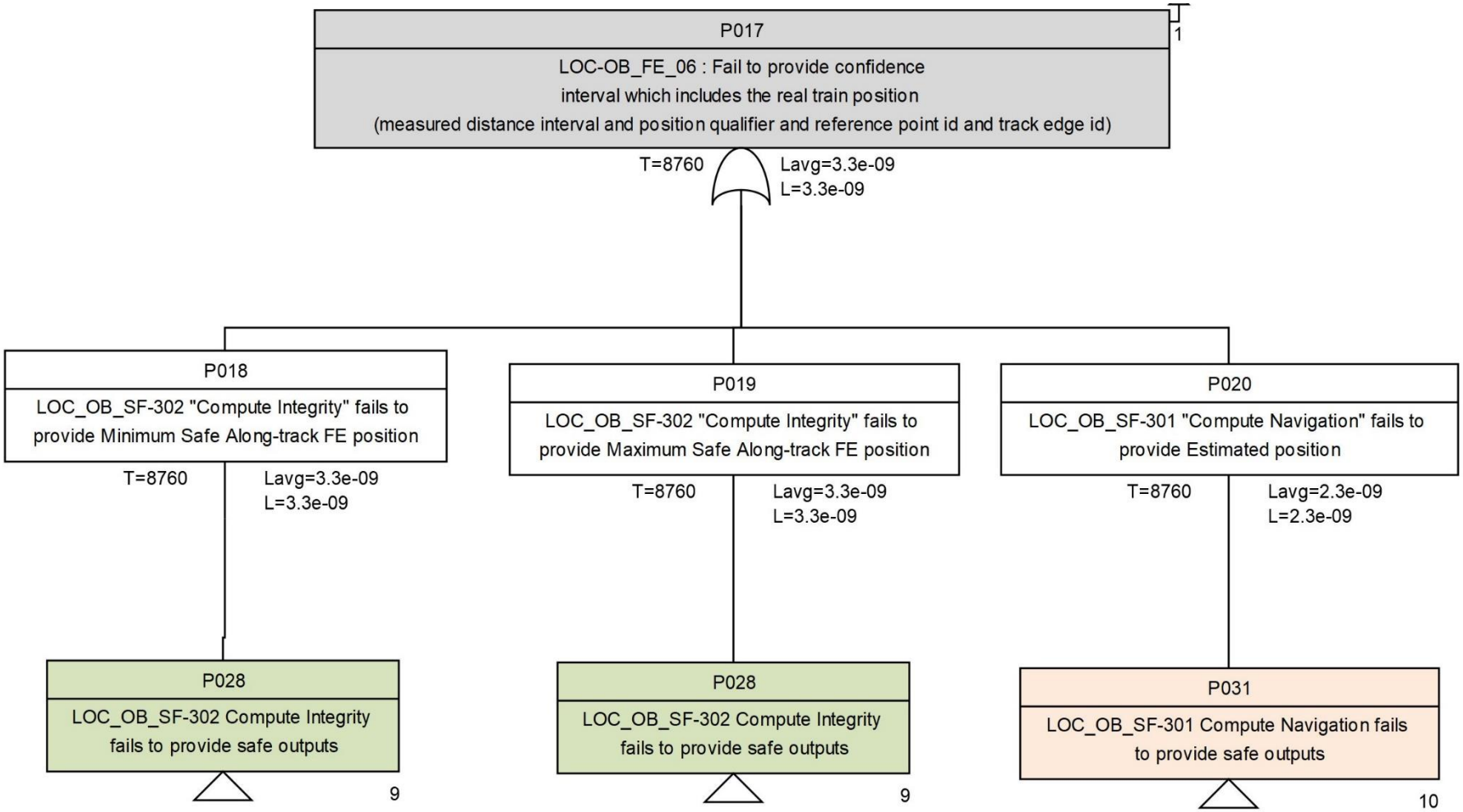

Figure 12: Fault-Tree for LOC_OB_FE_06 according to high-level architecture (D2.3)

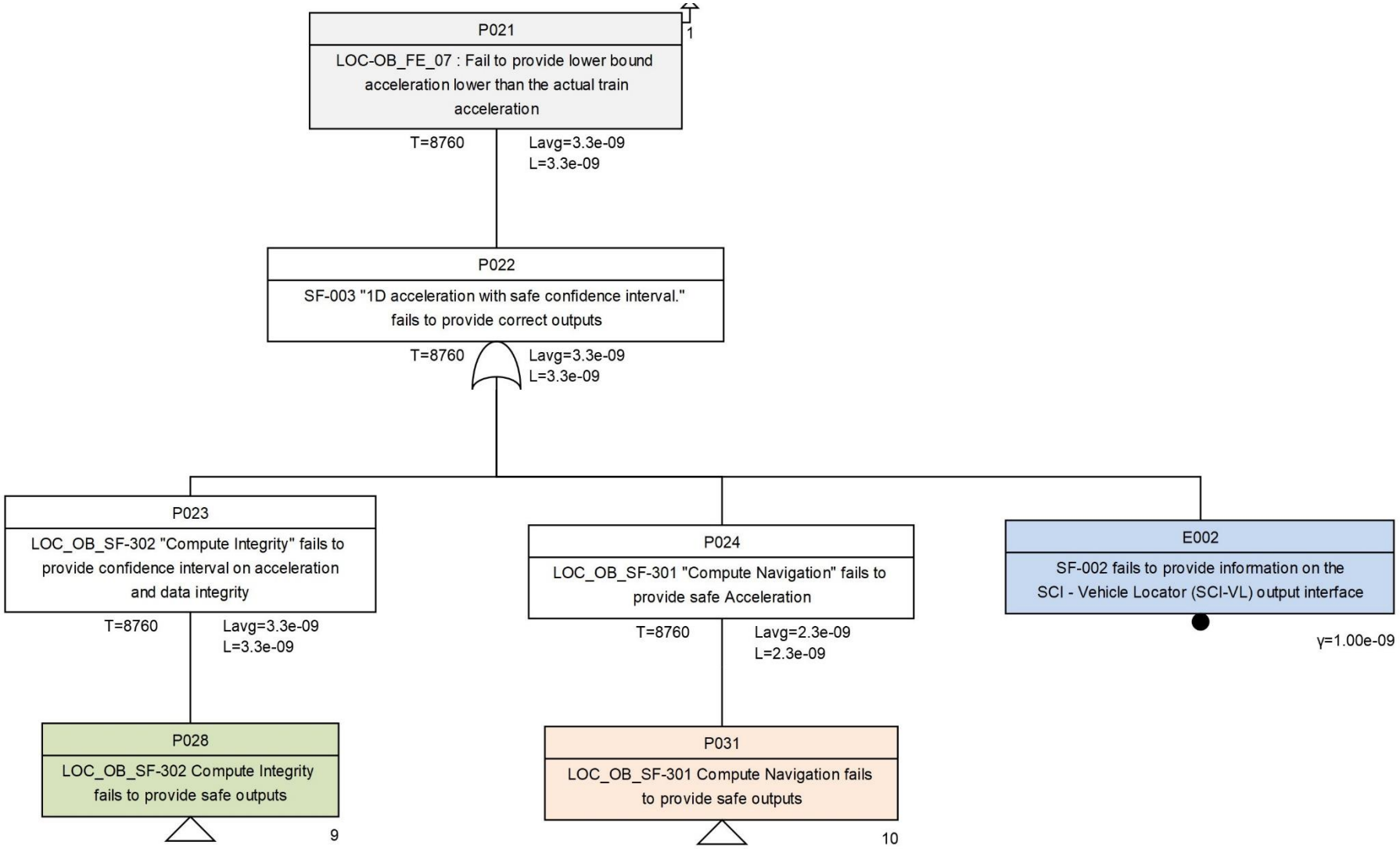

Figure 13: Fault-Tree for LOC_OB_FE_07 according to high-level architecture (D2.3)

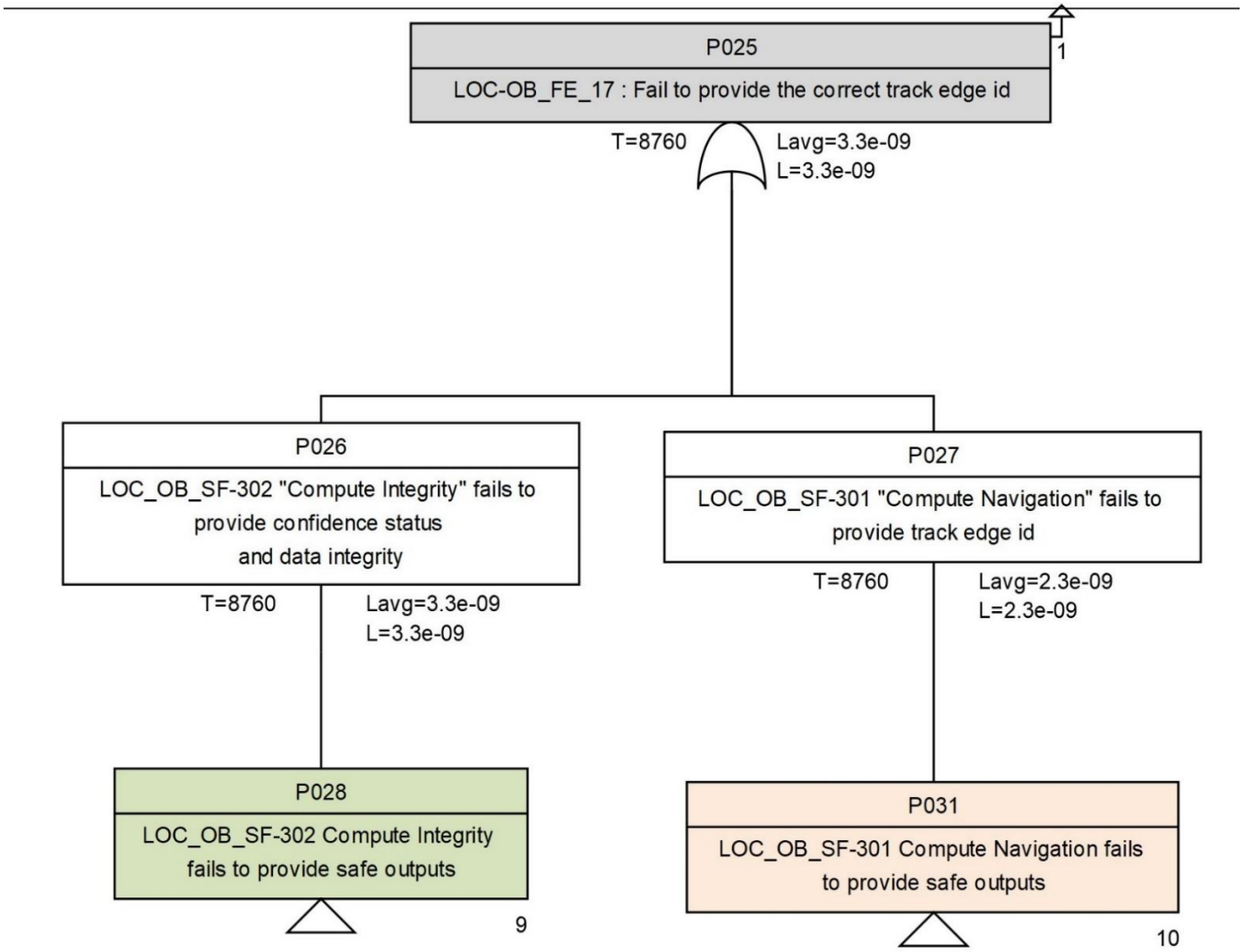

Figure 14: Fault-Tree for LOC_OB_FE_17 according to high-level architecture (D2.3)

4.2.3 Common functional blocks

Some assumptions are made in regards of the internal functional architecture described in D2.3 (see Figure 1):

Assumption 1: The integrity module uses the outputs from the Navigation module.

Assumption 2: It is assumed that only 3 of the 4 possible measurement systems is used.

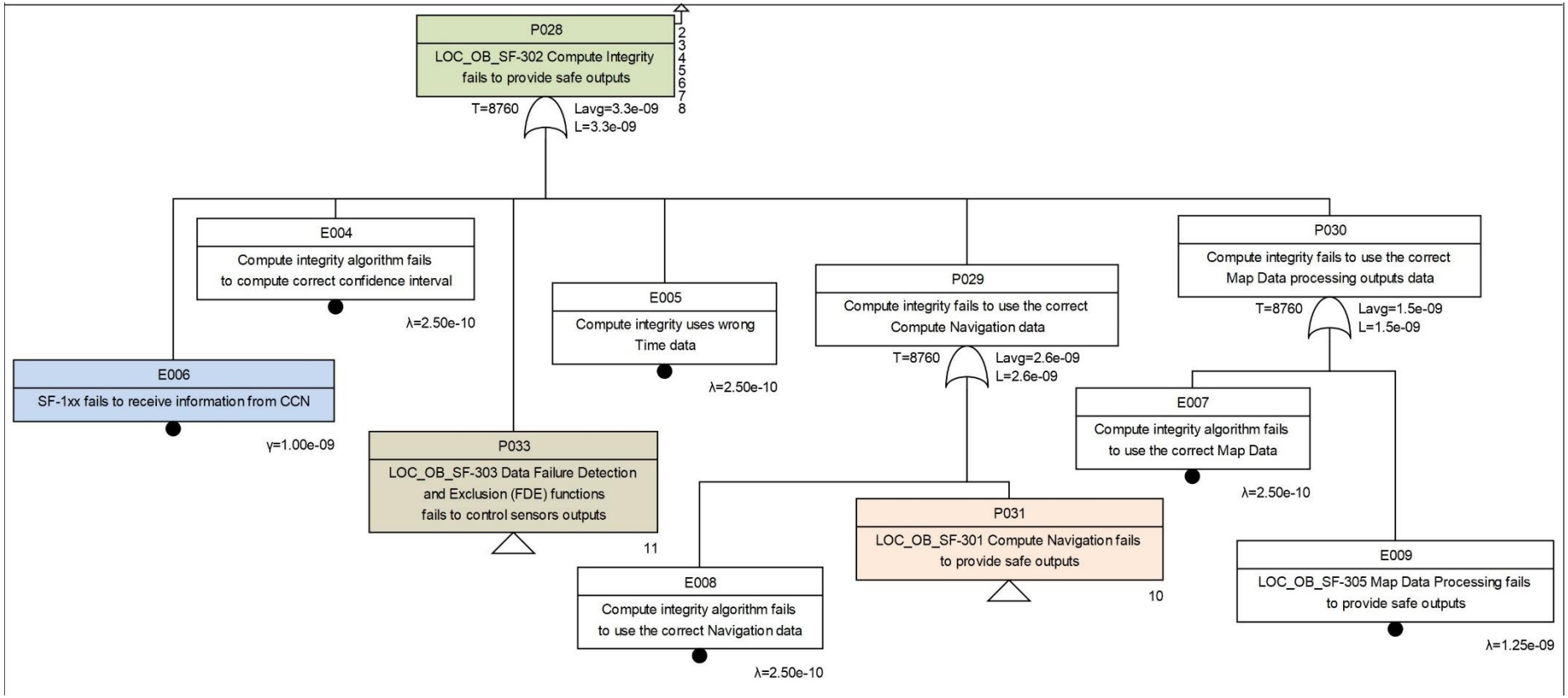

Figure 15: Fault-Tree for LOC_OB_SF_302 Compute integrity according to high-level architecture (D2.3)

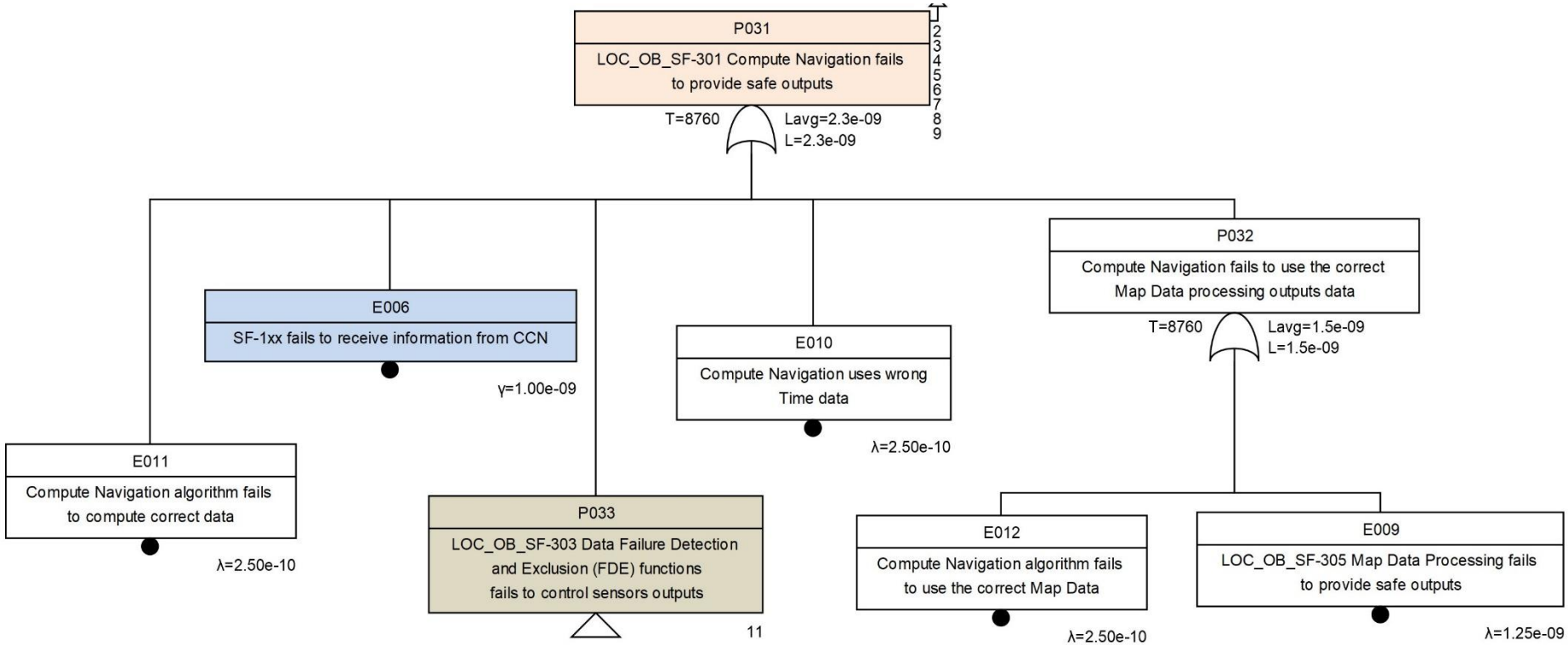

Figure 16: Fault-Tree for LOC_OB_SF_301 Compute Navigation according to high-level architecture (D2.3)

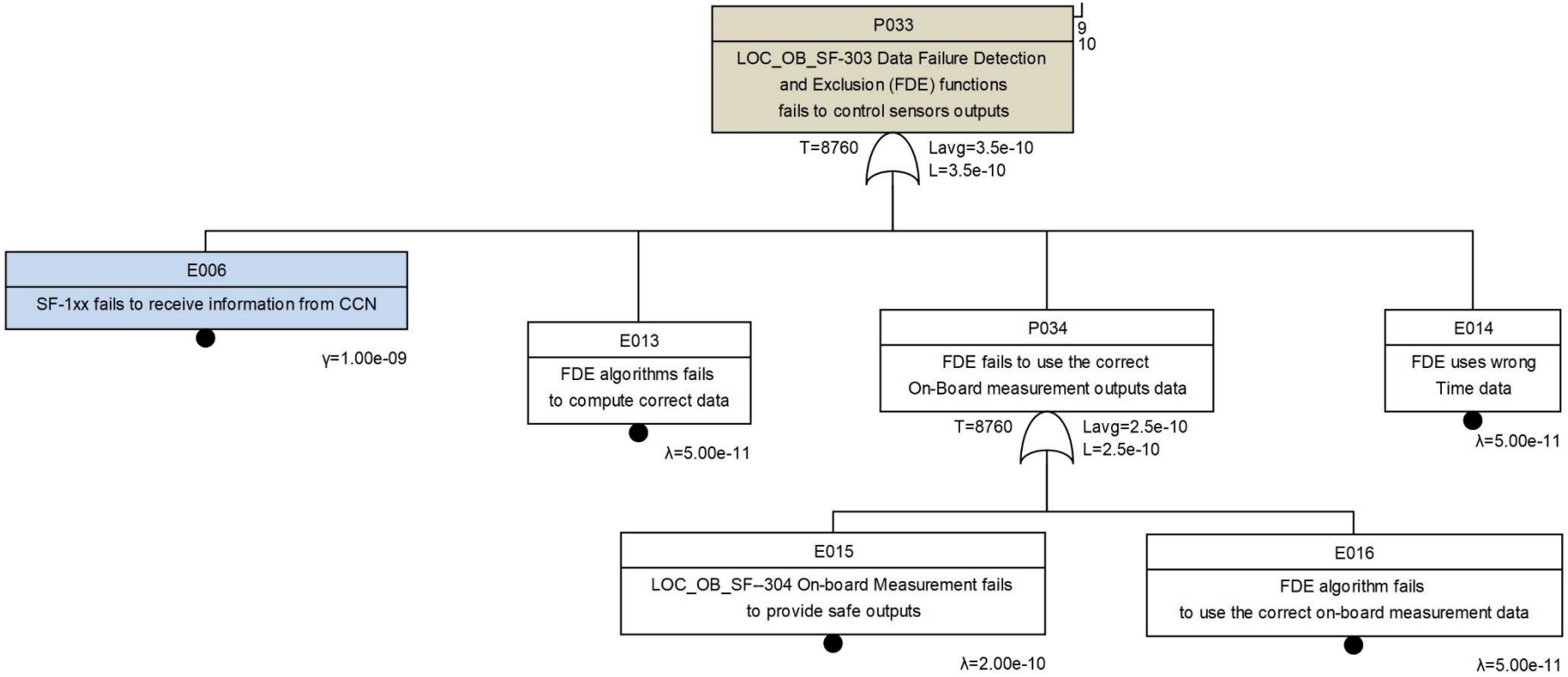

Figure 17: Fault-Tree for LOC_OB_SF_303 FDE according to high-level architecture (D2.3)

4.3 Fault-Tree: based on detailed system definition from D4.1

This section gives a refinement of the fault trees given in § 4.2 according to the detailed functional architecture from D4.1 (see § 2.3). Qualitative and quantitative analyses will be given in sections 5.2.2 and 5.3.2.

A set of open points (initialized from the analysis of the information from CLUG 1.0 D3.1.5 [R7]) are given for each fault tree.

4.3.1 Root gate

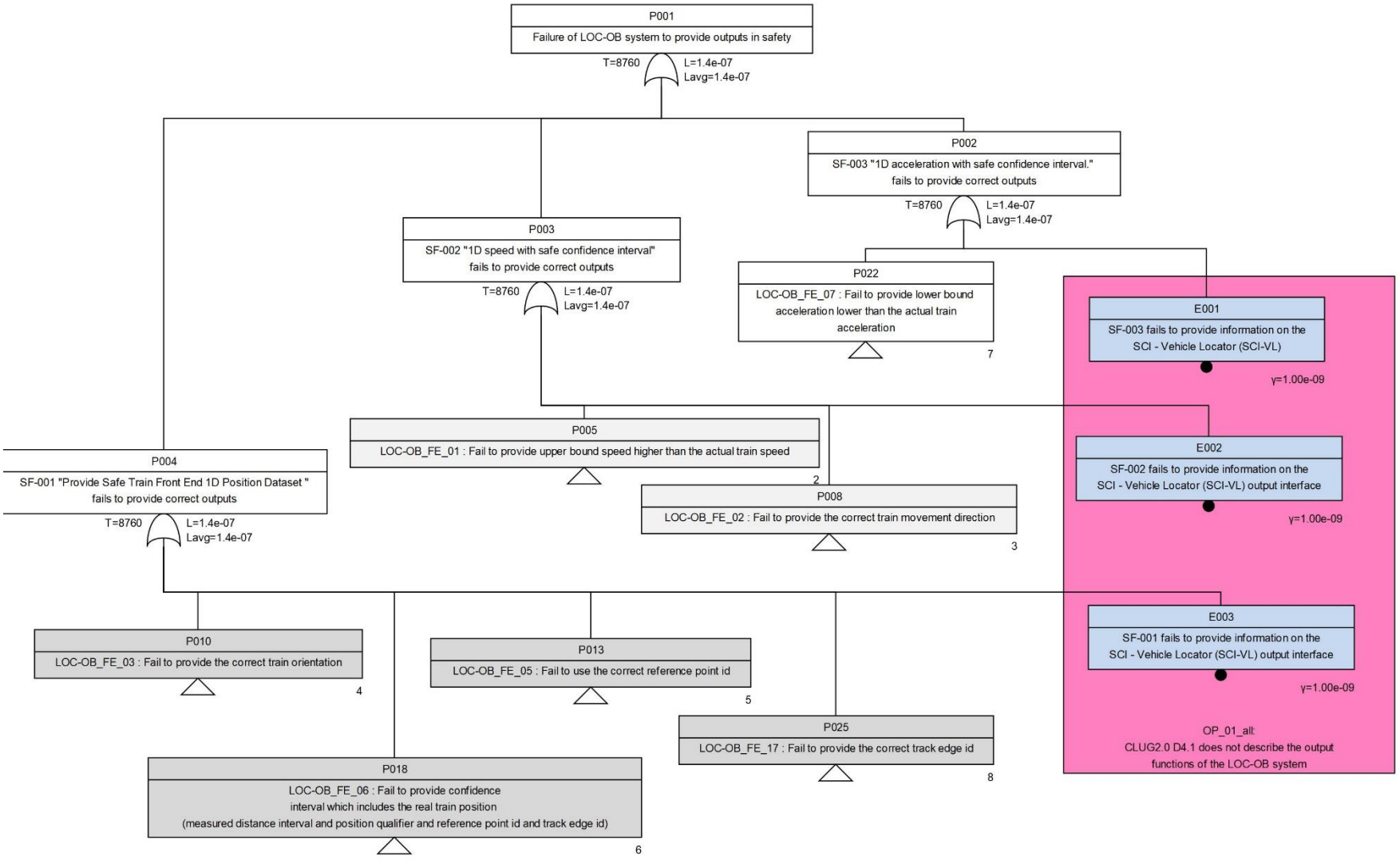

Figure 18: Fault-Tree for LOC-OB odometry function according to detailed architecture (D4.1)

4.3.2 Feared events

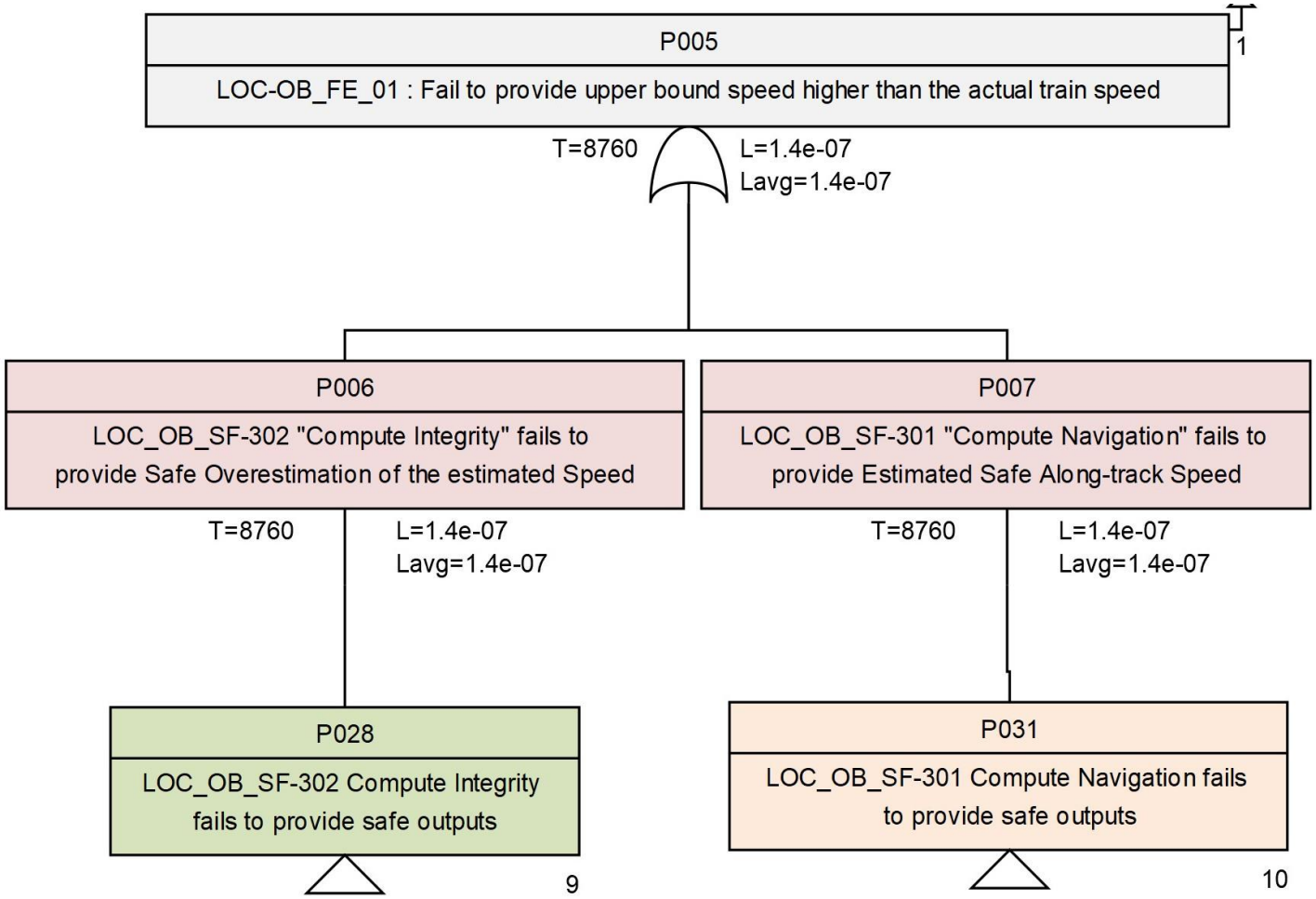

Figure 19: Fault-Tree for LOC_OB_FE_01 according to detailed architecture (D4.1)

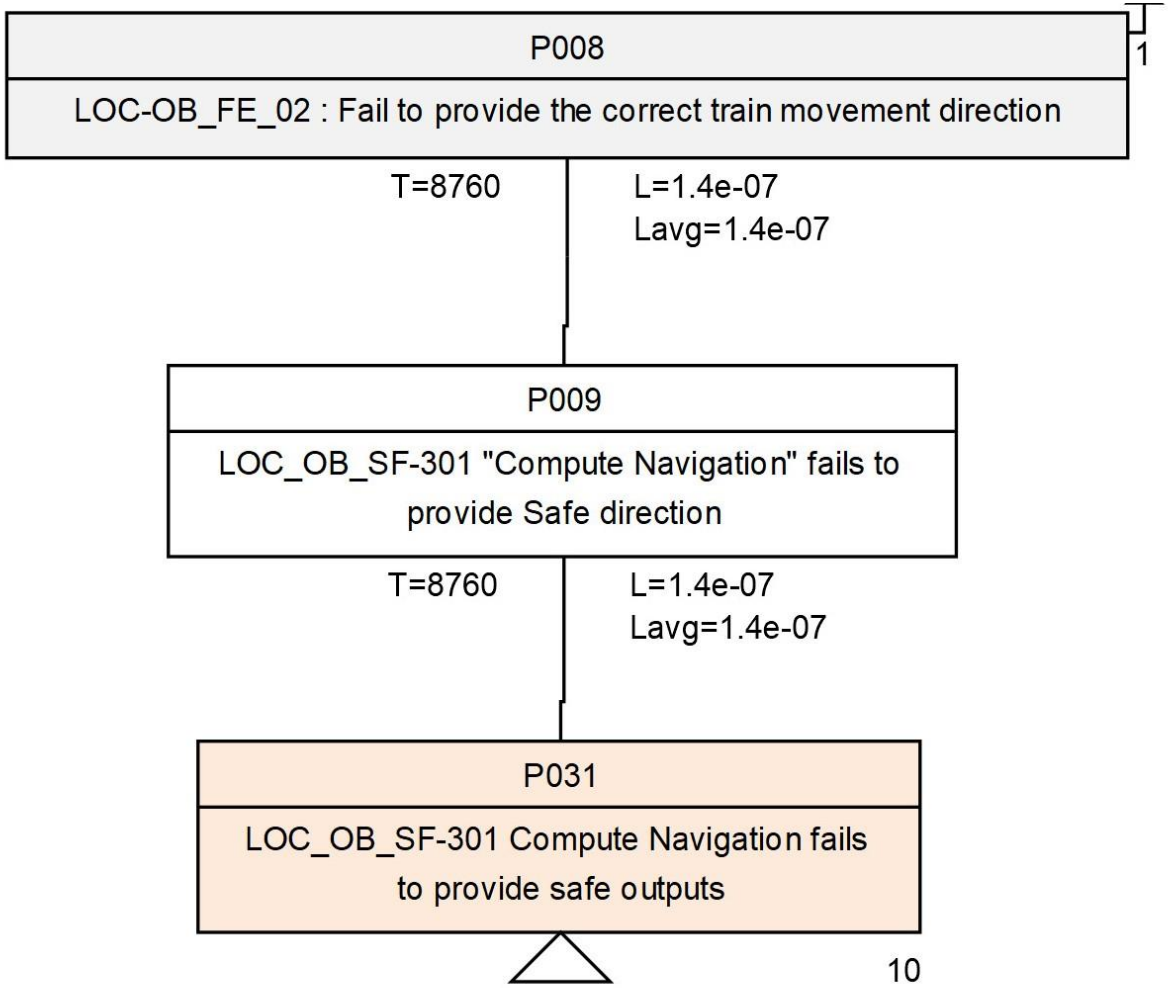

Figure 20: Fault-Tree for LOC_OB_FE_02 according to detailed architecture (D4.1)

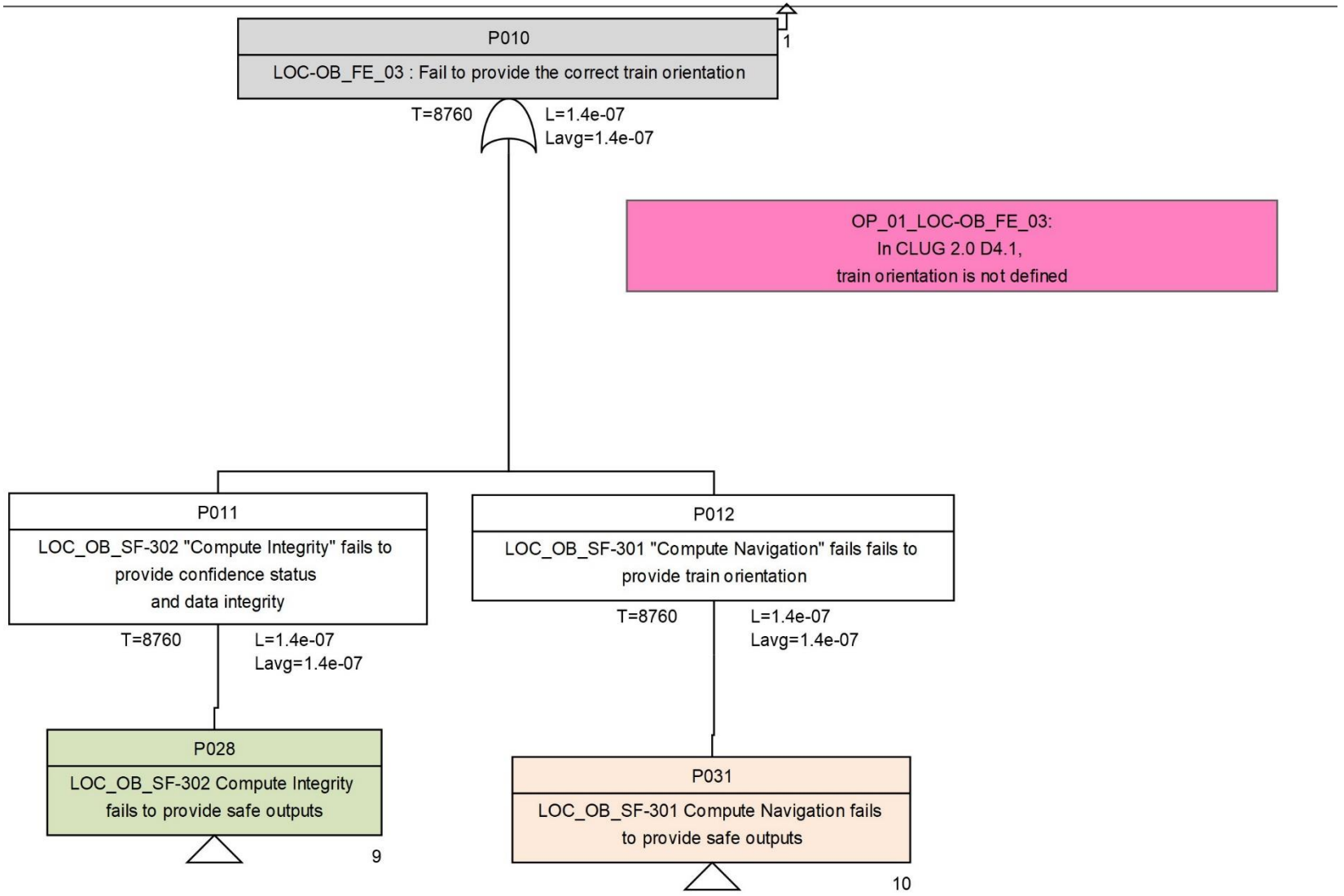

Figure 21: Fault-Tree for LOC_OB_FE_03 according to detailed architecture (D4.1)

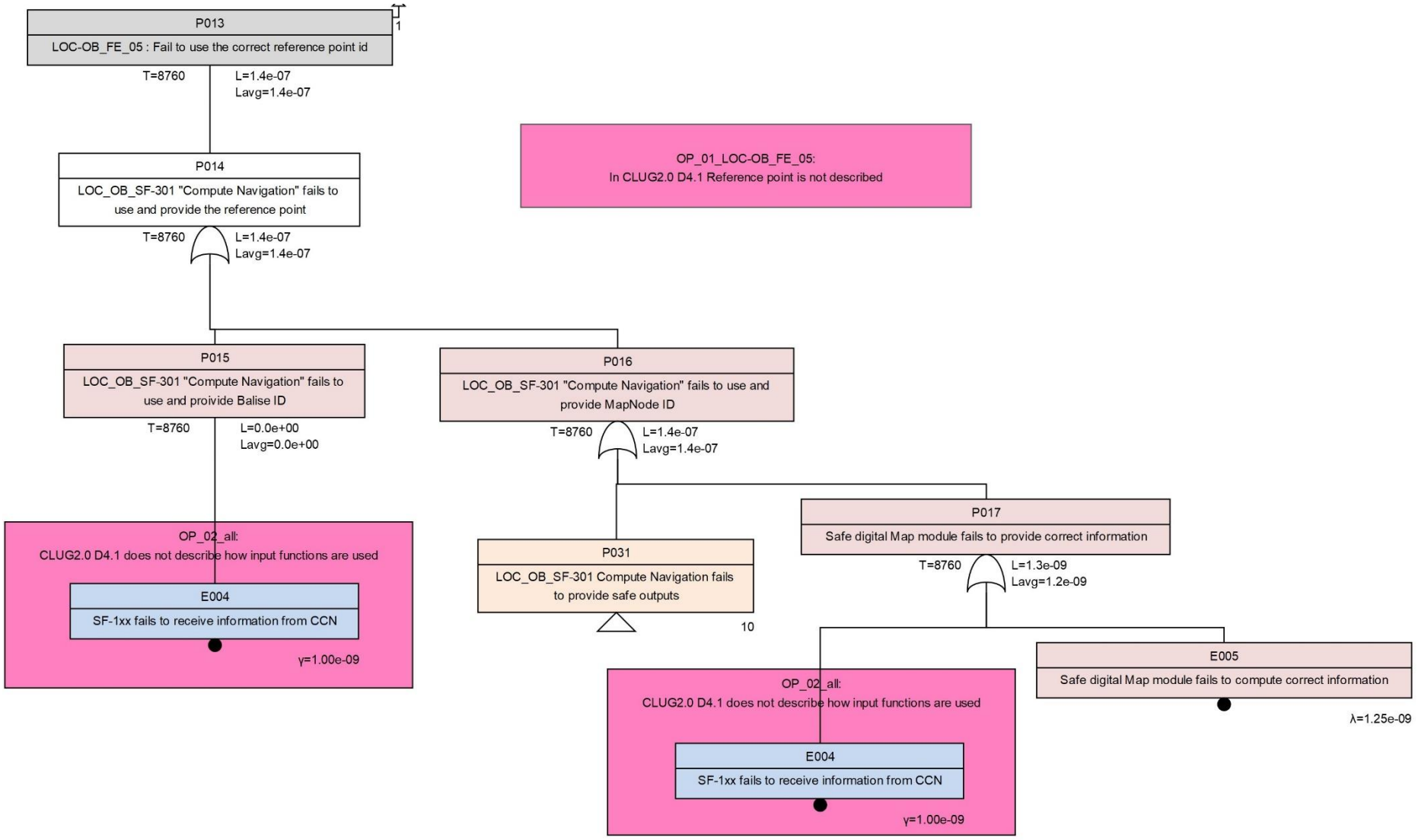

Figure 22: Fault-Tree for LOC_OB_FE_05 according to detailed architecture (D4.1)

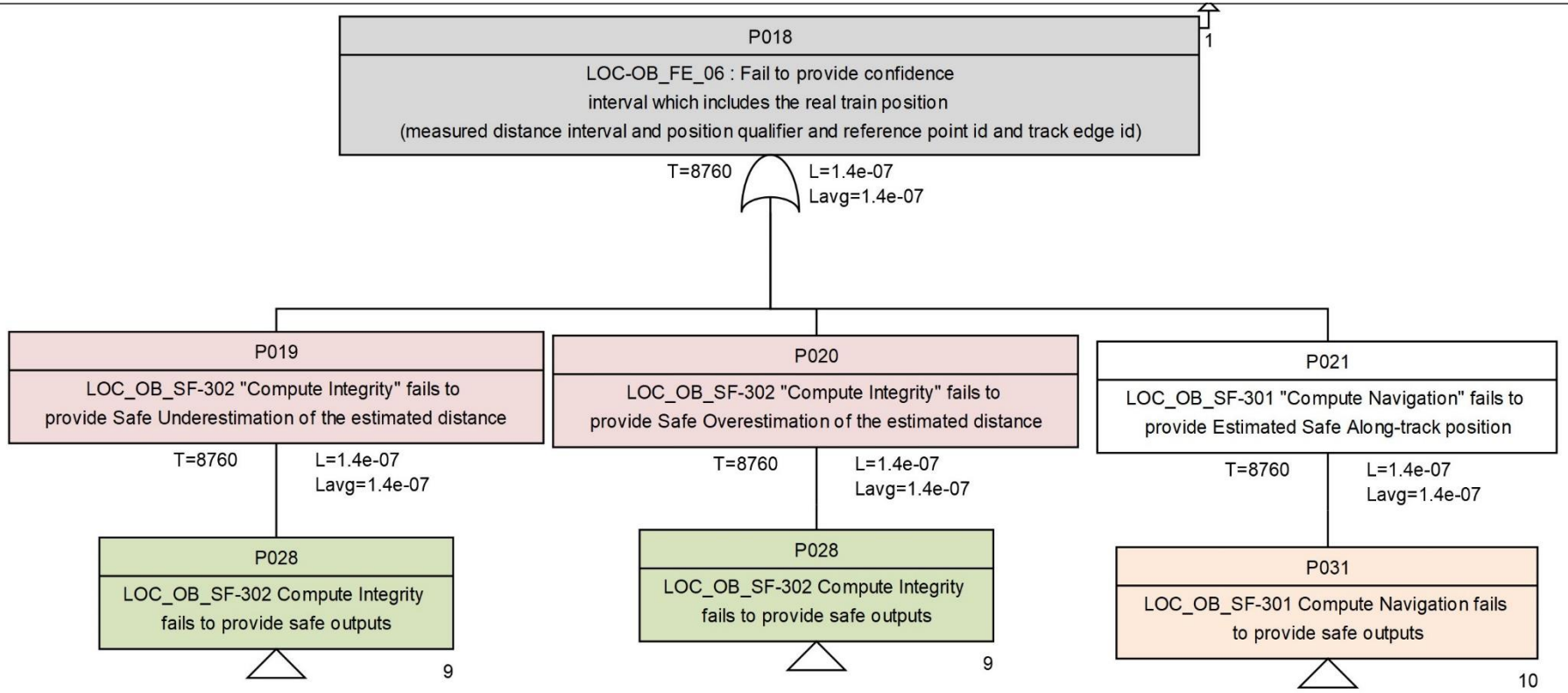

Figure 23: Fault-Tree for LOC_OB_FE_06 according to detailed architecture (D4.1)

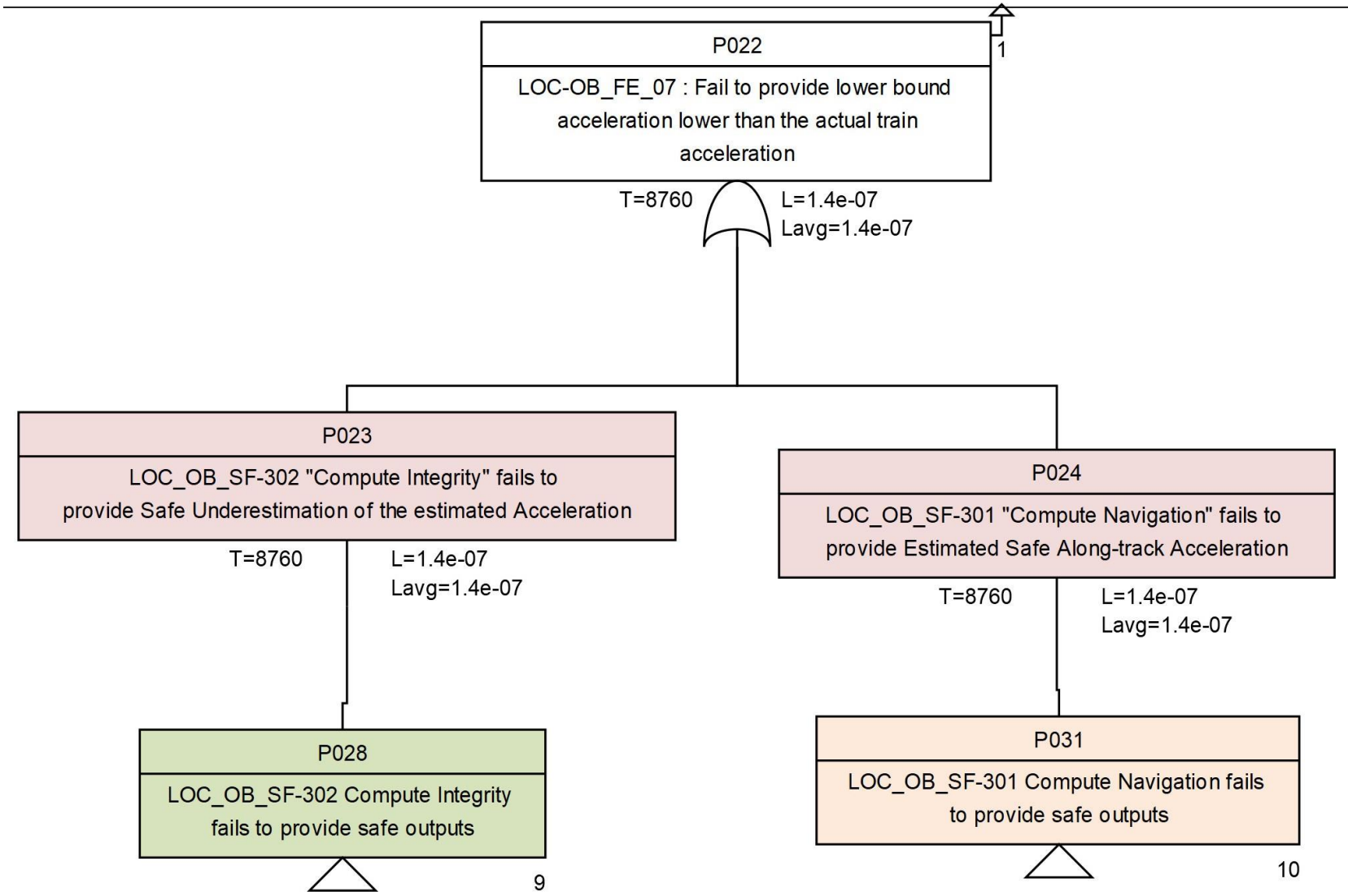

Figure 24: Fault-Tree for LOC_OB_FE_07 according to detailed architecture (D4.1)

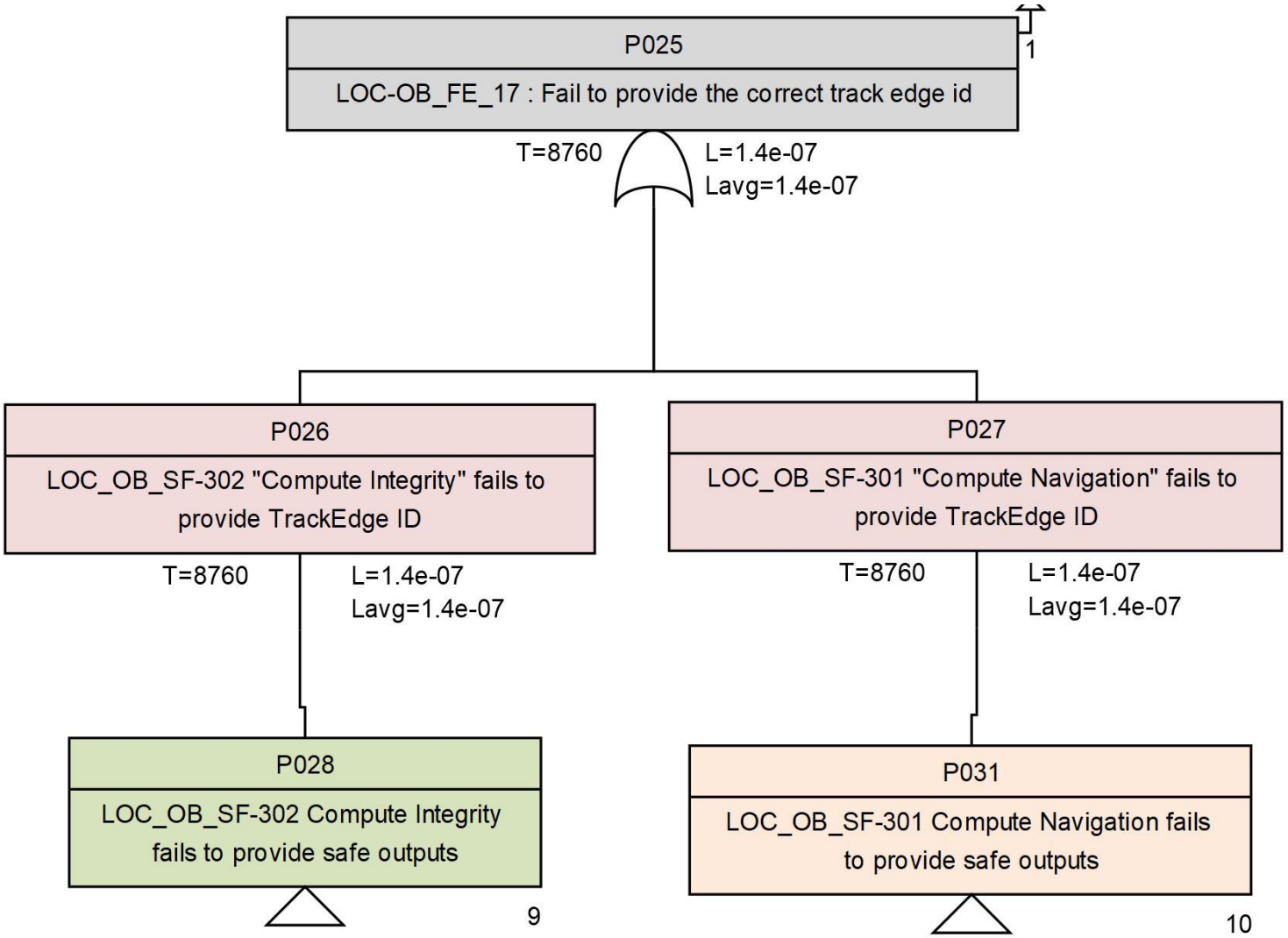

Figure 25: Fault-Tree for LOC_OB_FE_17 according to detailed architecture (D4.1)

4.3.3 Common functional blocks

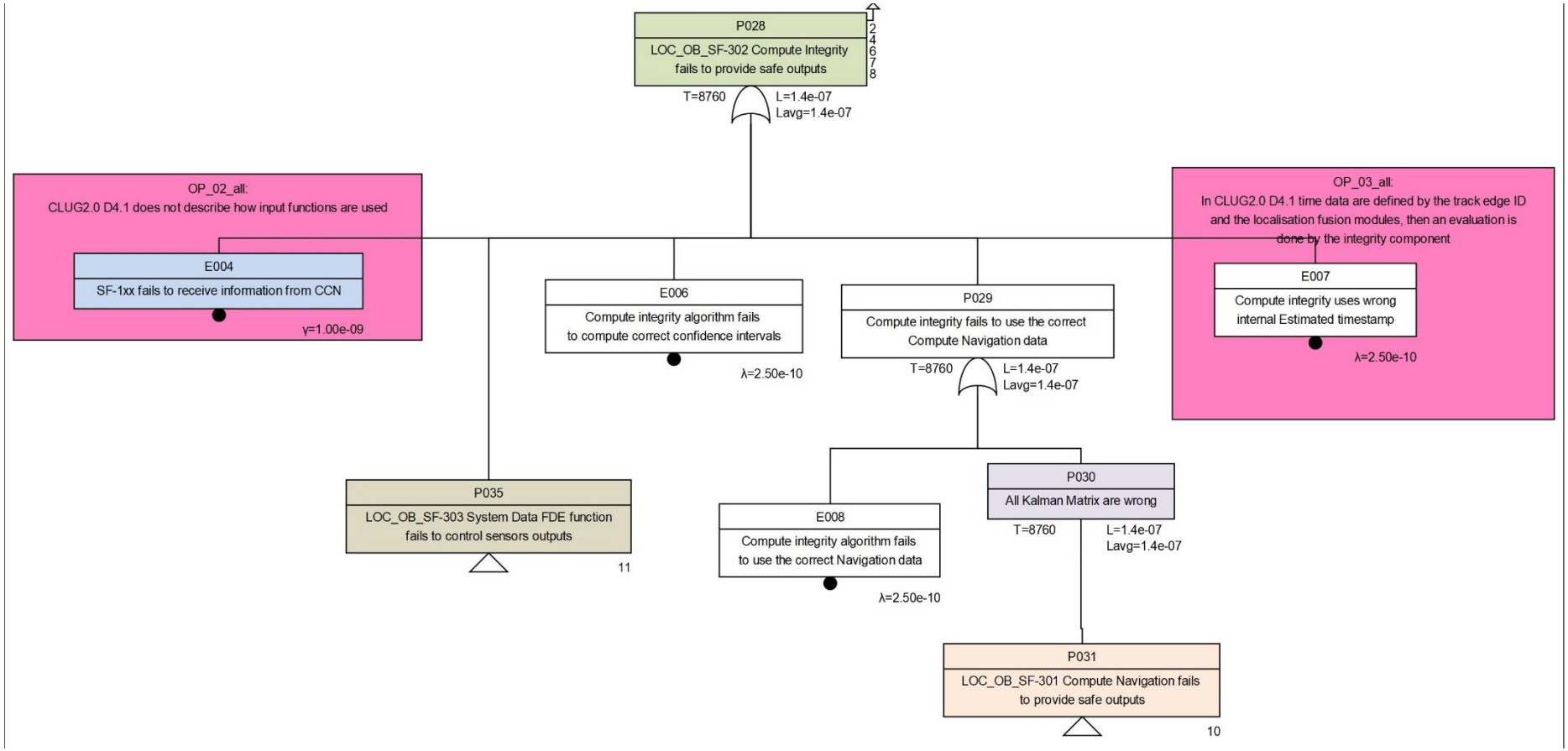

Figure 26: Fault-Tree for Compute integrity according to detailed architecture (D4.1)

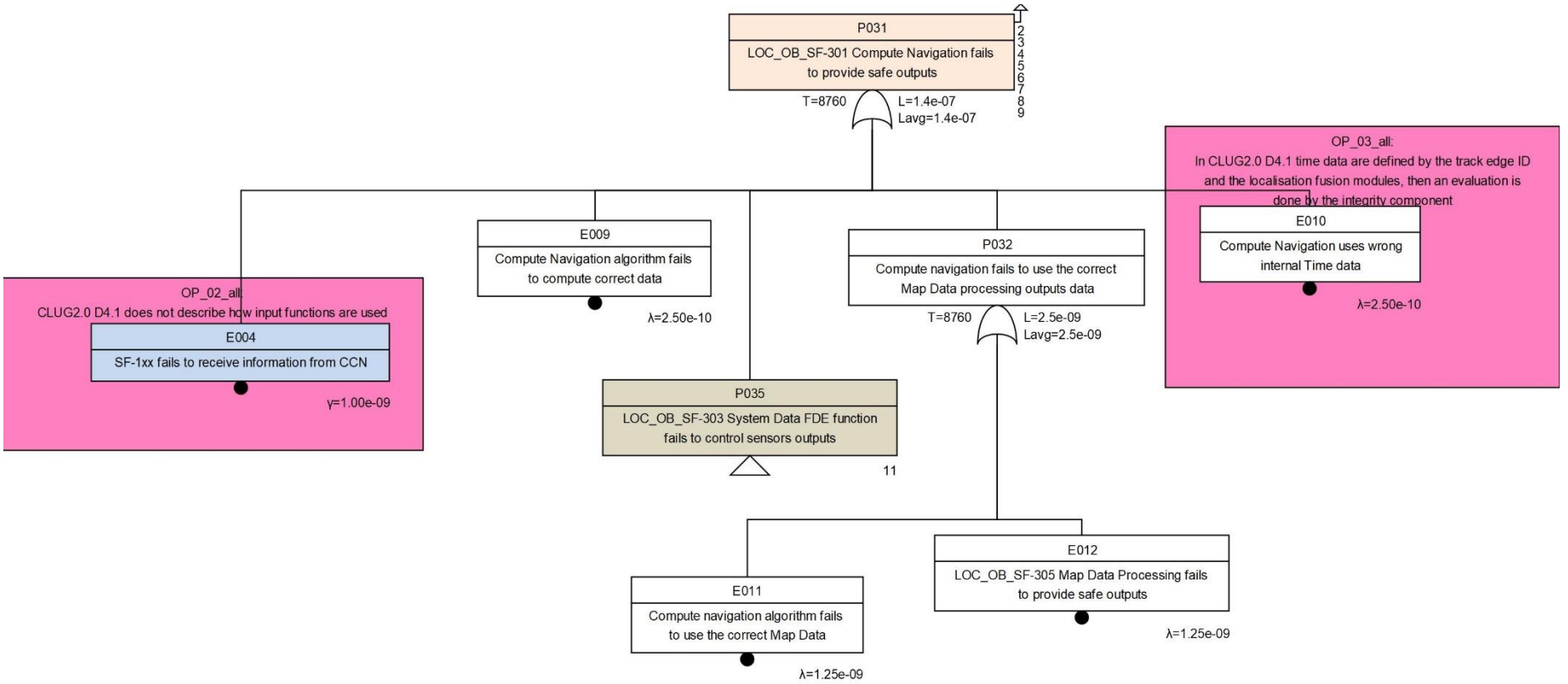

Figure 27: Fault-Tree for Compute Navigation according to detailed architecture (D4.1)

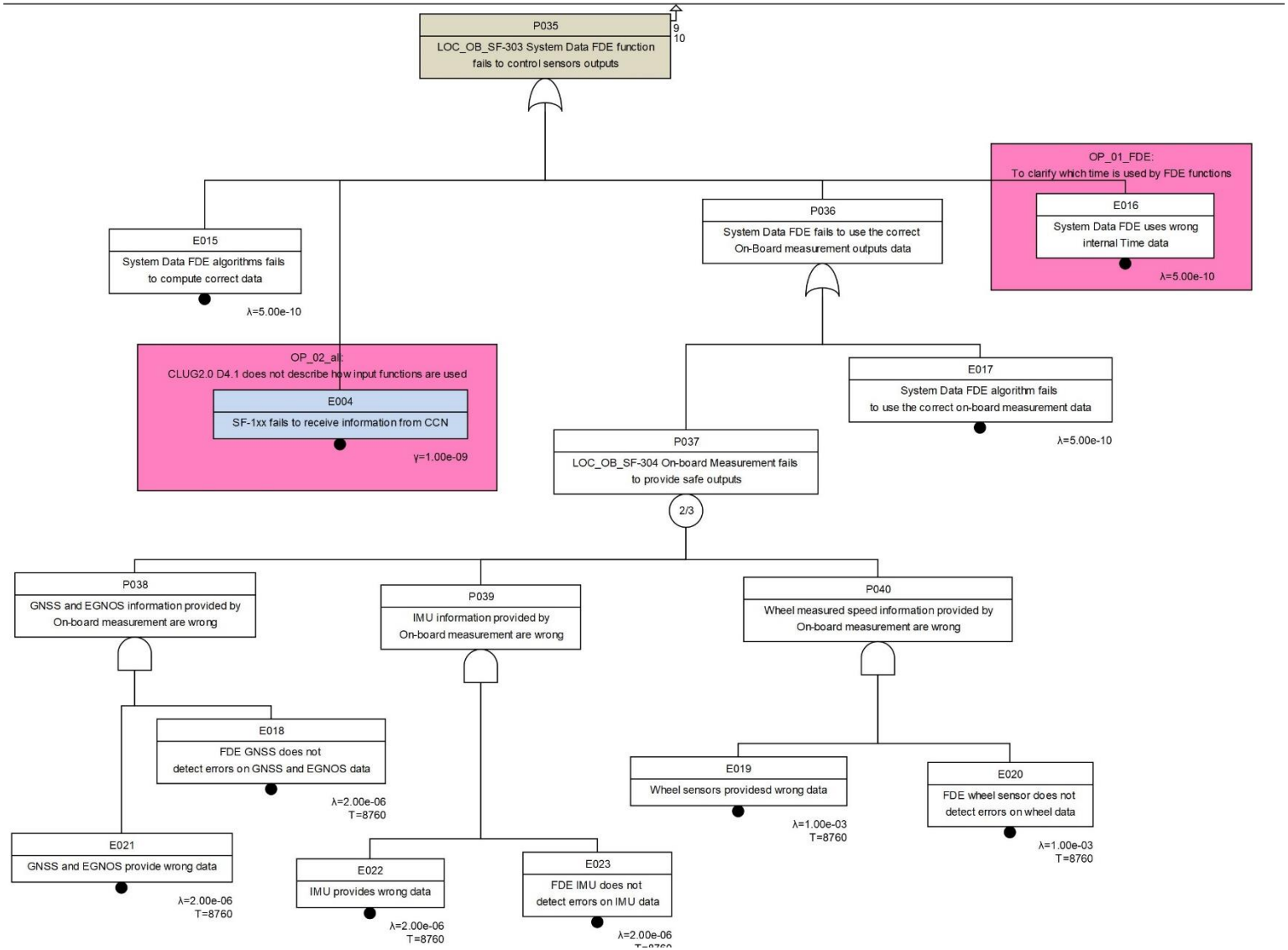

Figure 28: Fault-Tree for FDE according to detailed architecture (D4.1)

4.3.4 Open issues and assumptions

This section sum-up the list of open issues defined in §4.3.1 and §4.3.3 and to be covered in the final version of this document. This list has been initialized from the analysis of the information from CLUG 1.0 D3.1.5 [R7].

Identifier	Description	Status	Comments
OP_01_all	CLUG2.0 D4.1 does not describe the output functions of the LOC-OB system	Updated	
OP_02_all	CLUG2.0 D4.1 does not describe how input functions are used	Updated	
OP_03_all	In CLUG2.0 D4.1 time data are defined by the track edge ID and the localisation fusion modules, then an evaluation is done by the integrity component	Updated	
OP_01_LOC-OB_FE_01	*In CLUG1.0 D3.1.5 that SF-002 is not defined, it corresponds to Output 1a: Speed Min/max safe Confidence interval. Protocol interfaces are not described*	Closed	In D4.1, it corresponds to "Safe Overestimation of the estimated Speed" output of the integrity function
OP_01_LOC-OB_FE_02	*In CLUG1.0 D3.1.5, that SF-002 is not defined, it corresponds to Output 1a: Direction. Protocol interfaces are not described*	Closed	In D4.1, it corresponds to "Safe Direction" output of the navigation function
OP_02_LOC-OB_FE_02	*In CLUG1.0 D3.1.5 Compute Integrity part is not defined for the movement direction*	Closed	In D4.1, it corresponds to "Safe Direction" output of the navigation function
OP_01_LOC-OB_FE_03	In CLUG2.0 D4.1 Train orientation is not defined	Updated	
OP_03_LOC-OB_FE_03	*In CLUG2.0 D4.1 Compute Integrity part and Compute Navigation part are not defined for the train orientation*	Closed	Covered by OP_01_LOC-OB_FE_03
OP_01_LOC-OB_FE_05	In CLUG2.0 D4.1 Reference point is not described	Updated	
OP_02_LOC-OB_FE_05	*In CLUG2.0 D4.1 Compute Integrity is not defined for the reference point*	Closed	In D4.1, it corresponds to "Balise ID" and "MapNodeId" output of the navigation function
OP_01_LOC-OB_FE_17	*To clarify why the Compute integrity provides Safe Track Edge Status It seems from CLUG1.0 D3.1.5 that the safe track edge information is provided only by Navigation*	Closed	In D4.1, Track Edge ID is provided by the core navigation function and check by the integrity function
OP_01_Compute_Integrity	*In CLUG1.0 D3.1.5 parameters are defined by the FDE modules*	Closed	In D4.1, FDE system provides inputs to Control integrity

Identifier	Description	Status	Comments
OP_02_Compute_Integrity	*In CLUG1.0 D3.1.5 it is mainly Kalman matrix used from the navigation*	Closed	Confirmed in D4.1
OP_03_Compute_Integrity	*In CLUG1.0 D3.1.5 .5 Map data are not directly used by compute integrity*	Closed	Confirmed in D4.1
OP_01_FDE	To clarify which time is used and provided by FDE functions, as the one provided by GNSS is not the one shared with the users or other inputs	Updated	
OP_01_Track_Edge:	*Loop between function Navigation and Track_Edge to clarify*	Closed	Simplification for the D4.1 version

Table 9: Open issues related to the Fault tree analysis

4.4 Alternative architecture solution analysis

This section gives the upper-level analysis of the dual chain according to the proposals of D4.1 and §2.4. Qualitative and quantitative analyses will be given in sections 5.2.3 and 5.3.3.

Event E002 is related to the top gate of the first chain as described in § 1.1 Event E003 is not detailed in this analysis.

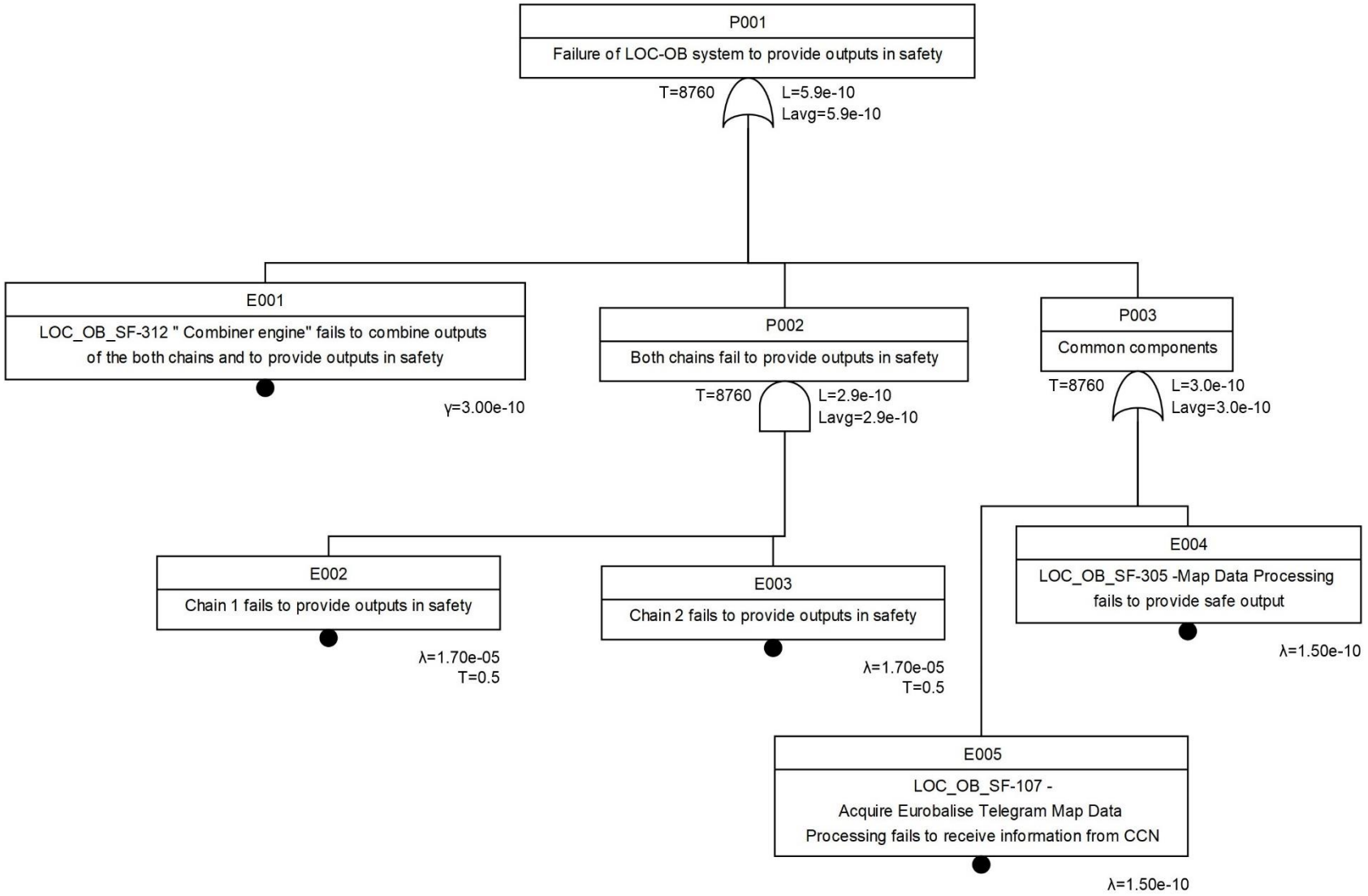

Figure 29: Fault-Tree for the dual chain according to alternative architecture (D4.1)

5 FUNCTIONAL ARCHITECTURE ANALYSES

5.1 Methods

The analysis performed in this to document aims to identify the role of the input information to produce safe output functions according to the fault trees defined in the previous section.

The analysis takes, in each case, as top gate the root gates (see §4.2.1 Figure 7 and § 4.3.1 Figure 18) which synthesis the feared events identified in §3.1.

The first method chosen is a qualitative approach commonly called “minimal cut set evaluation” which allow to evaluate the minimal combination of events which cause the failure of the top event of the tree. Minimal cut sets are a list of minimal (necessary and sufficient) component failed states which cause the system failure mode.

The second method chosen is a quantitative approach which allow to evaluate the system failure rate according to the failure rate for each component. The probability law use for this analysis is the exponential law dedicated to unrepairable components. Thus, the probability of failure of the top event of a tree is computed following the exponential law: $Q(t) = 1 - e^{-\lambda t}$.

For the quantification of the tree, the worst case is considered with a use of the system 24h by day, 365 days by year, i.e., 8760 hours. It is assumed that all the failure are detected continuously.

5.2 Qualitative analysis

5.2.1 High level system definition from D2.3

The table below contains all the min cuts of order 1: the failure of each event of this table shall lead to the failure of the root event of the system.

Event	Description	Comments
E001	SF-003 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	See **RA-RAMS-FTA-02**
E002	SF-002 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	See **RA-RAMS-FTA-02**
E003	SF-001 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	See **RA-RAMS-FTA-02**
E004	Compute integrity algorithm fails to compute correct confidence interval	See **RA-App-Archi_1-02** and **RA-App-Archi_2-02** and **RA-RAMS-03**
E005	Compute integrity uses wrong Time data	See **RA-RAMS-FTA-01** and **RA-RAMS-FTA-04**

Event	Description	Comments
E006	SF-1xx fails to receive information from CCN	See **RA-RAMS-FTA-03**
E007	Compute integrity algorithm fails to use the correct Map Data	See **RA-App-Archi_1-02** and **RA-App-Archi_2-02**
E008	Compute integrity algorithm fails to use the correct Navigation data	See **RA-App-Archi_1-02** and **RA-App-Archi_2-02**
E009	LOC_OB_SF_305 Map Data Processing fails to provide safe outputs	See **RA-App-Archi_1-05** and **RA-App-Archi_2-05**
E010	Compute Navigation uses wrong Time data	See **RA-RAMS-FTA-01** and **RA-RAMS-FTA-04**
E011	Compute Navigation algorithm fails to compute correct data	See **RA-App-Archi_1-01** and **RA-App-Archi_2-01**
E012	Compute navigation algorithm fails to use the correct Map Data	See **RA-App-Archi_1-01** and **RA-App-Archi_2-01**
E013	System Data FDE algorithms fails to compute correct data	See **RA-App-Archi_1-03** and **RA-App-Archi_2-03**
E014	FDE uses wrong Time data	See **RA-RAMS-FTA-01** and **RA-RAMS-FTA-04**
E015	LOC_OB_SF_304 On-board Measurement fails to provide safe outputs	See **RA-App-Archi_1-04** and **RA-App-Archi_2-04**
E016	FDE algorithm fails to use the correct on-board measurement data	See **RA-App-Archi_1-03** and **RA-App-Archi_2-03**

Table 10: Minimal cut of order 1 from High Level analysis

From the description of the architecture made in D2.3 (see Figure 1):

- It is confirmed that the output functions shall provide information according to a safe protocol, see **RA-RAMS-FTA-02**.
- The exchange of information shall be made according on a common time management compatible with interoperability, see **RA-RAMS-FTA-01** and **RA-RAMS-FTA-04**.
- Input function shall acquire information according to a safe protocol, see **RA-RAMS-FTA-03**, however the architecture description does not allow to identify which input information is mandatory. Failure of an input function SF-1XX covered both the cases of omission and incorrect processing of the acquisition function.
- Each function is considered as a black-box, meaning that a failure of one of these function leads directly to the failure of the LOC-OB system: the safety requirements related to the apportionment in section 3 are confirmed (see Table 18 and Table 19).

5.2.2 Detailed system definition from D4.1

The table below contains all the min cuts of order 1: the failure of each event of this table shall lead to the failure of the root event of the system.

Event	Description	Comments
E001	SF-003 fails to provide information on the SCI - Vehicle Locator (SCI-VL)	See **RA-RAMS-FTA-02**
E002	SF-002 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	See **RA-RAMS-FTA-02**
E003	SF-001 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	See **RA-RAMS-FTA-02**
E004	SF-1xx fails to receive information from CCN	See **RA-RAMS-FTA-03**
E005	Safe digital Map module fails to compute correct information	See **RA-App-Archi_1-05** and **RA-App-Archi_2-05**
E006	Compute integrity algorithm fails to compute correct confidence intervals	See **RA-App-Archi_1-02** and **RA-App-Archi_2-02** and **RA-RAMS-03**
E007	Compute integrity uses wrong internal Estimated timestamp	See **RA-App-Archi_1-02** and **RA-App-Archi_2-02**
E008	Compute integrity algorithm fails to use the correct Navigation data	See **RA-App-Archi_1-02** and **RA-App-Archi_2-02**
E009	Compute Navigation algorithm fails to compute correct data	See **RA-App-Archi_1-01** and **RA-App-Archi_2-01**
E010	Compute Navigation uses wrong internal Time data	See **RA-RAMS-FTA-01** and **RA-RAMS-FTA-04**
E011	Compute navigation algorithm fails to use the correct Map Data	See **RA-App-Archi_1-01** and **RA-App-Archi_2-01**
E012	LOC_OB_SF_305 Map Data Processing fails to provide safe outputs	See **RA-App-Archi_1-05** and **RA-App-Archi_2-05**
E015	System Data FDE algorithms fails to compute correct data	See **RA-App-Archi_1-03** and **RA-App-Archi_2-03**
E016	System Data FDE uses wrong internal Time data	See **RA-App-Archi_1-03** and **RA-App-Archi_2-03**
E017	System Data FDE algorithm fails to use the correct on-board measurement data	See **RA-App-Archi_1-03** and **RA-App-Archi_2-03**

Table 11: Minimal cut of order 1 according to detailed architecture (D4.1)

Four others minimal cut sets have been calculated at order 4: the failure of the 4 events of each set shall lead to the failure of the root event of the system.

Event	Description	Comments
E018	FDE GNSS does not detect errors on GNSS and EGNOS data	
E019	Wheel sensors provides wrong data	
E020	FDE wheel sensor does not detect errors on wheel data	

Event	Description	Comments
E021	GNSS and EGNOS provide wrong data	
E019	Wheel sensors provide wrong data	
E020	FDE wheel sensor does not detect errors on wheel data	
E022	IMU provides wrong data	
E023	FDE IMU does not detect errors on IMU data	
E018	FDE GNSS does not detect errors on GNSS and EGNOS data	
E021	GNSS and EGNOS provide wrong data	
E022	IMU provides wrong	
E023	FDE IMU does not detect errors on IMU data	

Table 12: Minimal cut of order 2 according to detailed architecture (D4.1)

From the description of the architecture made in CLUG 1.0 (see Figure 2):

- It is confirmed that the output functions shall provide information according to a safe protocol, see **RA-RAMS-FTA-02**.
- The exchange of information shall be according on a common time management compatible with interoperability, see **RA-RAMS-FTA-01** and **RA-RAMS-FTA-04**.
- Input function shall acquire information according to a safe protocol, see **RA-RAMS-FTA-03**, however the architecture description does not allow to identify which input information is mandatory.
- Each function is considered as a black-box, meaning that a failure of one of these function leads directly to the failure of the LOC-OB system: the safety requirements related to the apportionment in section 3 are confirmed (see Table 18 and Table 19).
- All the kind of measurement sensors proposed in D2.3 are not used at the same time. A combination of only 3 sets of sensors or two sets of sensors will be considered in the proposed solutions of D4.1. It is considered that at least two different kinds of measurement (on three if there are three sets in input) are needed as input of the LOC-OB at a given time. Assuming that failure rates of the GNSS measurement and the IMU measurement are 2.00e-6 per hour (see **LOC-OB-FTA-Ass-27** and **LOC-OB-FTA-Ass-28**), and the wheel sensor measurement is 1.00e-3 per hour (see **LOC-OB-FTA-Ass-29**), the probability to have a failure of the LOC-OB system is higher in the case of wheel sensors are used. Solutions including at least GNSS and IMU solution available at any time shall be preferred to reach the safety target expected on the outputs.

5.2.3 Alternative architecture solution analysis from D4.1

The table below contains all the min cuts of order 1: the failure of each event of this table shall lead to the failure of the root event of the system.

Event	Description	Comments
E001	LOC_OB_SF-312 " Combiner engine" fails to combine outputs of the both chains and to provide outputs in safety	See **RA-App-Archi_2-08** and **RA-App-Archi_2-09**
E004	LOC_OB_SF-305 -Map Data Processing fails to provide safe output	See **RA-App-Archi_1-05** and **RA-App-Archi_2-05**
E005	LOC_OB_SF-107 - Acquire Eurobalise Telegram Map Data Processing fails to receive information from CCN	See **RA-App-Archi_1-06** and **RA-App-Archi_2-06**

Table 13: Minimal cut of order 1 according to alternative architecture (D4.1)

One minimal cut set has been calculated at order 2: the failure of the 2 events of the set shall lead to the failure of the root event of the system.

Event	Description	Comments
E002	Chain 1 fails to provide outputs in safety	See **RA-App-Archi_2-07** and **RA-App-Archi_2-10**
E003	Chain 2 fails to provide outputs in safety	

Table 14: Minimal cut of order 2 according to alternative architecture (D4.1)

From the description of the alternative dual chain architecture made in D4.1 (see Figure 3):

- The combiner function shall be designed to detect and isolate the failure of each chain and shall be designed to avoid introduction of new faults: the safety requirements related to the apportionment in section 3 are confirmed (see **RA-App-Archi_2-08** and **RA-App-Archi_2-09**).
- Each chain shall provide outputs in safety : the safety requirements related to the apportionment in section 3 are confirmed (see **RA-App-Archi_2-07** and **RA-App-Archi_2-10**)..
- Independence between the two chains shall be ensured (see **RA-App-Archi_2-10**).

5.3 Quantitative analysis

5.3.1 High level system definition from D2.3

With the strong hypotheses given in Table 15, based on apportionment on Archi_1 (see Table 18), the failure rate for the LOC-OB odometry system is calculated on the trees given in § 4.2:

Lambda system: 3.35e-09

This confirmed the apportionment defined in section 3.3.1 on Archi_1 to reach the safety target of the system is difficult to put adapt on a single chain architecture.

A finer description of the functional internal architecture is necessary to obtain more accurate results (see §5.3.2 and §5.3.3) and some assumptions shall be confirmed.

Event	Description	Law	Probability	Lambda	Comments/ coverage
E001	SF-003 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	Constant	1,00E-09		**RA-RAMS-FTA-01 RA-RAMS-FTA-02**
E002	SF-002 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	Constant	1,00E-09		**RA-RAMS-FTA-01 RA-RAMS-FTA-02**
E003	SF-001 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	Constant	1,00E-09		**RA-RAMS-FTA-01 RA-RAMS-FTA-02**
E004	Compute integrity algorithm fails to compute correct confidence interval	Exponential		2.5e-10	**RA-App-Archi_1-02**
E005	Compute integrity uses wrong Time data	Exponential		2.5e-10	**RA-RAMS-FTA-04**
E006	SF-1xx fails to receive information from CCN	Constant	1,00E-09		**RA-RAMS-FTA-02 RA-RAMS-FTA-03**

Event	Description	Law	Probability	Lambda	Comments/ coverage
E007	Compute integrity algorithm fails to use the correct Map Data	Exponential		2.5e-10	**RA-App-Archi_1-02**
E008	Compute integrity algorithm fails to use the correct Navigation data	Exponential		2.5e-10	**RA-App-Archi_1-02**
E009	LOC_OB_SF-305 Map Data Processing fails to provide safe outputs	Exponential		1.25e-09	**RA-App-Archi_1-05**
E010	Compute Navigation uses wrong Time data	Exponential		2.5e-10	**RA-RAMS-FTA-04**
E011	Compute Navigation algorithm fails to compute correct data	Exponential		2.5e-10	**RA-App-Archi_1-01**
E012	Compute Navigation algorithm fails to use the correct Map Data	Exponential		2.5e-10	**RA-App-Archi_1-01**
E013	FDE algorithms fails to compute correct data	Exponential		5,00E-11	**RA-App-Archi_1-03**
E014	FDE uses wrong Time data	Exponential		5,00E-11	**RA-RAMS-FTA-04**
E015	LOC_OB_SF--304 On-board Measurement fails to provide safe outputs	Exponential		2,00E-10	**RA-App-Archi_1-04**
E016	FDE algorithm fails to use the correct on-board measurement data	Exponential		5,00E-11	**RA-App-Archi_1-03**

Table 15: Probability of failure or failure rate of the events for High Level analysis

5.3.2 Detailed system definition from D4.1

With the strong hypotheses given in Table 16, , based on apportionment on Archi_1 (see Table 18), the failure rate for the LOC-OB odometry system is calculated on the trees given in §1.1:

Lambda system: 1.43e-07

As more detailed is given for this architecture on the interaction between the components, the results is better (assumption of the failure rate of the whole measurement is better).

This confirmed the apportionment defined in section 3.3.2 for Archi_1 does not allow to reach the safety target with a single chain, under the assumptions defined.

However, the safety target allocated to one of the chains of the alternative architecture Archi_2 can be reached.

Besides the assumptions on the measurement modules and individual modules should be confirmed.

Event	Description	Law	Probability	Lambda	Comments/ coverage
E001	SF-003 fails to provide information on the SCI - Vehicle Locator (SCI-VL)	Constant	1,00E-09		**RA-RAMS-FTA-01 RA-RAMS-FTA-02**
E002	SF-002 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	Constant	1,00E-09		**RA-RAMS-FTA-01 RA-RAMS-FTA-02**
E003	SF-001 fails to provide information on the SCI - Vehicle Locator (SCI-VL) output interface	Constant	1,00E-09		**RA-RAMS-FTA-01 RA-RAMS-FTA-02**
E004	SF-1xx fails to receive information from CCN	Constant	1,00E-09		**RA-RAMS-FTA-01 RA-RAMS-FTA-03**
E005	Safe digital Map module fails to compute correct information	Exponential		1.25e-09	**RA-App-Archi_1-05**
E006	Compute integrity algorithm fails to compute correct confidence intervals	Exponential		2.5e-10	**RA-RAMS-FTA-02 RA-RAMS-FTA-03**
E007	Compute integrity uses wrong internal Estimated timestamp	Exponential		2.5e-10	**RA-RAMS-FTA-04**

Event	Description	Law	Probability	Lambda	Comments/ coverage
E008	Compute integrity algorithm fails to use the correct Navigation data	Exponential		2.5e-10	**RA-App-Archi_1-02**
E009	Compute Navigation algorithm fails to compute correct data	Exponential		2.5e-10	**RA-App-Archi_1-01**
E010	Compute Navigation uses wrong internal Time data	Exponential		2.5e-10	**RA-RAMS-FTA-04**
E011	Compute navigation algorithm fails to use the correct Map Data	Exponential		1.25e-09	**RA-App-Archi_1-05**
E012	LOC_OB_SF-305 Map Data Processing fails to provide safe outputs	Exponential		1.25e-09	**RA-App-Archi_1-05**
E015	System Data FDE algorithms fails to compute correct data	Exponential		5,00E-10	**RA-App-Archi_1-03**
E016	System Data FDE uses wrong internal Time data	Exponential		5,00E-10	**RA-RAMS-FTA-04**
E017	System Data FDE algorithm fails to use the correct on-board measurement data	Exponential		5,00E-10	**RA-App-Archi_1-03**
E018	FDE GNSS does not detect errors on GNSS and EGNOS data	Exponential		2,00E-06	**LOC-OB-FTA-Ass-27**
E019	Wheel sensors provide wrong data	Exponential		1,00E-03	**LOC-OB-FTA-Ass-29**
E020	FDE wheel sensor does not detect errors on wheel data	Exponential		1,00E-03	**LOC-OB-FTA-Ass-29**
E021	GNSS and EGNOS provide wrong data	Exponential		2,00E-06	**LOC-OB-FTA-Ass-27**

Event	Description	Law	Probability	Lambda	Comments/ coverage
E022	IMU provides wrong data	Exponential		2,00E-06	**LOC-OB-FTA-Ass-28**
E023	FDE IMU does not detect errors on IMU data	Exponential		2,00E-06	**LOC-OB-FTA-Ass-28**

Table 16: Probability of failure or failure rate of the events for detailed architecture (D4.1)

5.3.3 Alternative architecture solution analysis from D4.1

With the strong hypotheses given in Table 17, based on the apportionment on Archi_2, the failure rate for the LOC-OB odometry system is calculated on the trees given in §1.1:

Lambda system: 5.89e-10

This confirmed that in the case of a dual chain architecture, the safety target of the system can be reached under the conditions summarized in section 6.

Besides the assumptions on the measurement modules and individual modules should be confirmed.

Event	Description	Law	Probability	Lambda	Comments/ coverage
E001	LOC_OB_SF-312 " Combiner engine" fails to combine outputs of both chains and to provide outputs in safety	Constant	3,00E-10		**RA-App-Archi_2-08**
E002	Chain 1 fails to provide outputs in safety	Exponential		1.7e-05	
E003	Chain 2 fails to provide outputs in safety	Exponential		1.7e-05	**RA-App-Archi_2-07**
E004	LOC_OB_SF-305 -Map Data Processing fails to provide safe output	Exponential		1.5e-10	**RA-App-Archi_2-05**
E005	LOC_OB_SF-107 - Acquire Eurobalise Telegram Map Data Processing fails to receive information from CCN	Exponential		1.5e-10	**RA-App-Archi_2-06**

Table 17: Probability of failure or failure rate of the events for alternative architecture (D4.1)

6 SYNTHESIS RESULTS

6.1 Safety requirements

This section gives the synthesis of the safety requirements identified in this document.

First the safety requirements related to the apportionment according Archi_1:

Id	RAMS requirements	Comments
RA-App-Archi_1-01	For Archi_1, the ***LOC_OB_SF-301*** Compute Navigation function shall be designed in SIL4 with a TFFR ≤ 0.25e-9 per hour on the output	
RA-App-Archi_1-02	For Archi_1, the ***LOC_OB_SF-302*** Compute Integrity function shall be designed in SIL4 with a TFFR ≤ 0.25e-9 per hour on the output	
RA-App-Archi_1-03	For Archi_1, the ***LOC_OB_SF-303*** System Data FDE function shall be designed in SIL4 with a TFFR ≤ 0.5e-10 per hour on the output	
RA-App-Archi_1-04	For Archi_1, the ***LOC_OB_SF-304*** On-Board Measurement function shall be designed in SIL4 with a TFFR ≤ 0.2e-9 per hour on the output	
RA-App-Archi_1-05	For Archi_1, the ***LOC_OB_SF-305*** Map Data Processing function shall be designed in SIL4 with a TFFR ≤ 0.125e-9 per hour on the output	
RA-App-Archi_1-06	For Archi_1, the **LOC_OB_SF-107** Acquire Eurobalise Telegram function shall be designed in SIL4 with a TFFR ≤ 0.125e-9 per hour on the output	This safety requirement is to discuss in regard to the system which provide the Eurobalise Telegram.

Table 18: List of safety requirements identified for apportionment on Archi_1

Second the safety requirements related to the apportionment according Archi_2:

Id	RAMS requirements	Comments
RA-App-Archi_2-01	For Archi_2, the ***LOC_OB_SF-301*** Compute Navigation function shall be designed in SIL2 with a TFFR ≤ 5.8e-6 per hour on the output	
RA-App-Archi_2-02	For Archi_2, the ***LOC_OB_SF-302*** Compute Integrity function shall be designed in SIL2 with a TFFR ≤ 5.8e-6 per hour on the output	

Id	RAMS requirements	Comments
RA-App-Archi_2-03	For Archi_2, the ***LOC_OB_SF-303*** System Data FDE function shall be designed in SIL2 with a TFFR ≤ 0.8e-6 per hour on the output	
RA-App-Archi_2-04	For Archi_2, the ***LOC_OB_SF-304*** On-Board Measurement function shall be designed in SIL2 with a TFFR ≤ 0.5e-5 per hour on the output	
RA-App-Archi_2-05	For Archi_2, the ***LOC_OB_SF-305*** Map Data Processing function shall be designed in SIL4 with a TFFR ≤ 0.15e-9 per hour on the output	
RA-App-Archi_2-06	For Archi_2, the **LOC_OB_SF-107** Acquire Eurobalise Telegram function shall be designed in SIL4 with a TFFR ≤ 0.15e-9 per hour on the output	This safety requirement is to discuss in regard to the system which provides the Eurobalise Telegram.
RA-App-Archi_2-07	For Archi_2, each chain shall provide safe output functions with TFFR ≤ 1.7e-5 per hour.	
RA-App-Archi_2-08	For Archi_2, the ***LOC_OB_SF-312*** Combiner function shall be designed in SIL4 with a TFFR ≤ 0.3e-9 per hour on the output	
RA-App-Archi_2-09	In case of two chains with two independent computation functions and two independent sets of sensors, a combiner function shall implement a mechanism to detect and manage the failure of each chain in a given limited time in view to provide safe outputs.	The time to detect the failure of each chain shall be identified in the specification of the combiner function. A value of half an hour has been used in this analysis which gives conservative value.
RA-App-Archi_2-10	The two chains shall be designed with independent computation functions and independent sets of sensors.	

Table 19: List of safety requirements identified for apportionment on Archi_2

Finally, the safety requirements related to the interface analysis and the FTA:

Id	RAMS requirements	Origin	Comments
RA-RAMS-FTA-01	LOC-OB, user equipment and provider equipment shall use data exchange mechanisms in accordance with the safety, security and interoperability requirements.	D2.4	See SpecSysReq[035] in [R10]
RA-RAMS-FTA-02	LOC-OB shall provide its dataset in compliance with the future TSI through the SCI - Vehicle Locator (SCI-VL) interface with a 1.0e-9 ≤ TFFR < 1.0e-8 per hour.	D2.4	See SpecSysReq[036] in [R10]
RA-RAMS-FTA-03	LOC-OB shall receive in safety data set from the neighbors' on-board system in compliance with the future TSI through the dedicated SCI interfaces with a 1.0e-9 ≤ TFFR < 1.0e-8 per hour.	D2.4	See SpecSysReq[037], SpecSysReq[038], SpecSysReq[039], SpecSysReq[040], SpecSysReq[041], SpecSysReq[042], SpecSysReq[043], SpecSysReq[044], SpecSysReq[045], SpecSysReq[046] in [R10]
RA-RAMS-FTA-04	LOC-OB shall embed a safe and secure mechanism to detect delays and time incoherencies a 1.0e-9 ≤ TFFR < 1.0e-8 per hour.	D2.4	See SpecSysReq[034] in [R10]
RA-RAMS-FTA-05	*Intentionally deleted*		*Transformed in* ***RA-App-Archi_2-08***
RA-RAMS-FTA-06	*Intentionally deleted*		*Transformed in* ***RA-App-Archi_2-09***
RA-RAMS-FTA-07	*Intentionally deleted*		*Transformed in* ***RA-App-Archi_2-07***
RA-RAMS-FTA-08	*Intentionally deleted*		*Transformed in* ***RA-App-Archi_2-10***

Table 20: List of safety requirements identified.

6.2 Assumptions

The table presents the synthesis of assumptions identified during this analysis of the LOC-OB.

Id	Assumptions	Comments
LOC-OB-FTA-Ass-22	*Intentionally deleted*	
LOC-OB-FTA-Ass-23	This function covered the specific processing on the map data made in the core of the LOC-OB system. Map Data is provided by the function LOC_OB_SF-101.	Already defined in [R13] See § 2.5, function ***LOC_OB_SF-305***
LOC-OB-FTA-Ass-24	In this document the measurement functions cover the sensors, measurement algorithms and Fault detection and exclusion for a given technology.	Already defined in [R13] See § 2.5, function ***LOC_OB_SF-306***, ***LOC_OB_SF-307*** and ***LOC_OB_SF-308***
LOC-OB-FTA-Ass-25	*Intentionally deleted*	
LOC-OB-FTA-Ass-26	*Intentionally deleted*	
LOC-OB-FTA-Ass-27	For analysis on the chain 1, it is assumed that the measurement information provided by the GNSS + EGNOS sensor as a failure of 2.00e-6 per hour and the FDE associated as a failure rate of 2.00e-6 per hour.	See § 5.2.2 and 5.3.2 (from information provided in [R14])
LOC-OB-FTA-Ass-28	For analysis on the chain 1, it is assumed that the measurement information provided by the IMU sensor as a failure of 2.00e-6 per hour and the FDE associated as a failure rate of 2.00e-6 per hour.	See § 5.2.2 and 5.3.2 (from information provided in [R14])
LOC-OB-FTA-Ass-29	For analysis on the chain 1, it is assumed that the measurement information provided by the wheel sensor as a failure of 1.00e-3 per hour and the FDE associated as a failure rate of 1.00e-3 per hour.	See § 5.2.2 and 5.3.2 (from information provided in [R14])
LOC-OB-FTA-Ass-30	For analysis on the chain 2, it is assumed that the second chain need balise information provided in safety by the balise reader	See § 2.4
LOC-OB-FTA-Ass-31	For apportionment and analysis on the dual chain architecture, and to cover **RA-RAMS-FTA-06,** it is assumed that the combiner function can detect and manage the failure of each chain in less the 30 minutes.	See § 3.3.2

Table 21: Assumptions

6.3 Open points

The following open points are remaining opened:

Open point #	Issue	Action	Status
LOC-OB-OP-20	*It is to clarify when this Initialisation function is called and the exact interaction with other functions.*	*Already defined in [R13]* *See § 2.5, function* ***LOC_OB_SF-309***	Closed: this details of function is not considered in the final version
Loc-OB-OP-21	*It is to clarify how this Track Edge ID determination function is linked with the Navigation module, as it seems there is a loop between the function.*	*Already defined in [R13]* *See § 2.5, function* ***LOC_OB_SF-310***	Closed: this details of function is not considered in the final version
OP_01_all	CLUG2.0 D4.1 does not describe the output functions of the LOC-OB system	Updated	
OP_02_LOC-OB_FE_02	*In CLUG1.0 D3.1.5 Compute Integrity part is not defined for the movement direction*	Closed	In D4.1, it corresponds to "Safe Direction" output of the navigation function
OP_01_LOC-OB_FE_03	In CLUG2.0 D4.1 Train orientation is not defined	Updated	
OP_03_LOC-OB_FE_03	*In CLUG2.0 D4.1 Compute Integrity part and Compute Navigation part are not defined for the train orientation*	Closed	Covered by OP_01_LOC-OB_FE_03
OP_01_LOC-OB_FE_05	In CLUG2.0 D4.1 Reference point is not described	Updated	
OP_02_LOC-OB_FE_05	*In CLUG2.0 D4.1 Compute Integrity is not defined for the reference point*	Closed	In D4.1, it corresponds to "Balise ID" and "MapNodeId" output of the navigation function
OP_01_LOC-OB_FE_17	*To clarify why the Compute integrity provides Safe Track Edge Status* *It seems from CLUG1.0 D3.1.5 that the safe track edge information is*	Closed	In D4.1, Track Edge ID is provided by the core navigation function and check by the integrity function

Open point #	Issue	Action	Status
	provided only by Navigation		
OP_01_Compute_Integrity	*In CLUG1.0 D3.1.5 parameters are defined by the FDE modules*	Closed	In D4.1, FDE system provides inputs to Control integrity
OP_02_Compute_Integrity	*In CLUG1.0 D3.1.5 it is mainly Kalman matrix used from the navigation*	Closed	Confirmed in D4.1
OP_03_Compute_Integrity	*In CLUG1.0 D3.1.5 .5 Map data are not directly used by compute integrity*	Closed	Confirmed in D4.1
OP_01_FDE	To clarify which time is used and provided by FDE functions, as the one provided by GNSS is not the one shared with the users or other inputs	Updated	
OP_01_Track_Edge:	*Loop between function Navigation and Track_Edge to clarify*	Closed	Simplification for the D4.1 version

Table 22: Open points

7 CONCLUSION

This document provides a high level external interfaces safety analysis but largely completed by a safety functional system analysis of the LOC-OB system, identifying how the exchanged between the LOC-OB system and its environment and the internal functional blocks shall be designed to cover the safety requirements already identified in the previous works, D2.1 [R8], D2.4 [R10] and D3.4 [R12]. This analysis provides too, via an apportionment approach, indications on the selection of safety targets expected on the input measurement means to achieve the TFFR expected on the output functions.

The analyses defined in this document are based on the description of the LOC-OB system given in D2.3 [R9] and in D4.1 [R14], in one case with only one chain of computation and in the second case with two chains and a combiner function.

From the apportionment and the qualitative and quantitative analyses, a set of requirements and assumptions are deduced which can be resumed to:

- External exchanges of information with the system using or providing information, according to SCI interfaces, shall be done with a TFFR =1.00e-9 per hour (for details see Table 20).
- Management of time-stamping of the information, shall be done in accordance to the user needs, with a TFFR =1.00e-9 per hour.
- In the context of an architecture with a single chain as defined in D.4.1 [R14] and § 2.3, the internal functions shall be designed in SIL4 with a TFFR < 1.0e-9 per hour on the output (for details see Table 18).
- In the context of an architecture with two chains as defined in D.4.1 [R14] and §2.4, the internal functions of each chain shall be designed in SIL2 with a TFFR < 1.0e-5 per hour on the output. The Combiner function shall be design in in SIL4 with a TFFR < 1.0e-9 per hour on the output and shall implement mechanisms to detect and isolate the faults between the chains (for details see Table 19).
- In the case of a dual chain architecture, both chains shall be designed with independent computation functions and independent sets of sensors and a combiner function shall be defined in safety to detect and isolate the faults of each chain. (see RA-App-Archi_2-09 and RA-App-Archi_2-10).

The conclusion on this analysis, is that under the assumptions given in § 6.2, the detailed architecture with one chain proposed cannot allow to reach the expected 1.0e-9 ≤ TFFR < 1.0e-8 per hour on the output functions. The alternative architecture with two chains (see § 2.4) allow to reach the expected safety objective 1.0e-9 ≤ TFFR < 1.0e-8 per hour on the output functions, if the safety requirements listed in §6.1 on the chains and the combiner function are fulfilled.

8 REFERENCE DOCUMENTS

[R1] [EN 50126-1-2017] Railway Applications - The Specification and demonstration of reliability, availability, maintainability and safety (RAMS) - Part 1: generic RAMS process

[R2] [EN 50126-2-2017] Railway Applications - The specification and demonstration of reliability, availability, maintainability and safety (RAMS) - Part 2: systems approach to safety

[R3] [EN 50129-2018] Railway applications - Communication, signalling and processing systems - Safety related electronic systems for signalling

[R4] [EN 50128-2011] Railway application – Communication, signalling and processing systems – Software for railway control and protection systems

[R5] [OCORA-TWS01-030] OCORA - System Architecture, v3.00, 30.11.2022

[R6] [OCORA-TWS01-035] OCORA - CCS-On-Board-(CCS-OB)-Architecture, v3.00, 30.11.2022

[R7] [CLUG1.0 – D3.1.5] TLOBU Solution A: Architecture and Design

[R8] [CLUG2.0 – D2.1] LOC-OB Operational Needs and System Capabilities of Localisation On-Board System

[R9] [CLUG2.0 – D2.3] LOC-OB System boundary, Architecture, and External interfaces (incl. DM)

[R10] [CLUG2.0 – D2.4] LOC-OB System Requirements

[R11] [CLUG2.0 – D3.1] LOC-OB System Context Analysis and RAMS Plan

[R12] [CLUG2.0 – D3.2] LOC-OB Preliminary Hazard Analysis

[R13] [CLUG2.0 – D3.3] LOC-OB System Failure Modes and Effects Analysis

[R14] [CLUG2.0 – D4.1] LOC-OB Functional System Architecture

[R15] [CLUG2.0 – D4.2] LOC-OB GNSS+EGNOS unit prototype including data FDE for LOC-OB design and description document

[R16] [CLUG2.0 – D4.3] LOC-OB Safe IMU sensor and data FDE for LOC-OB description document

[R17] [CLUG2.0 – D4.4] LOC-OB Speed sensor and data FDE for LOC-OB description document

[R18] [CLUG2.0 – D4.5] LOC-OB Eurobalise reader sensor and data FDE for LOC-OB description document

[R19] [CLUG2.0 – D4.6] LOC-OB Along track localization fusion algorithm design document

[R20] [CLUG2.0 – D4.7] LOC-OB Confidence Intervals computation & Integrity algorithm

[R21] [CLUG2.0 – D4.8] LOC-OB Track Selectivity Determination algorithm design document

[R22] [CLUG2.0 – D4.9] LOC-OB Start of Mission preliminary design

[R23] [CLUG2.0 – D4.10] LOC-OB On board Digital Map definition and interfaces

[R24] Tightly integrated map based train localization, Wenz and Ohrendorf-Weiss

[R25] Harmonized methodology for Safety Integrity Level allocation in a generic TCMS application, Ouedraogo and Beugin and El Koursi and Clarhaut and Renaux and Lisiecki, ESREL 2015

CLUG 2.0 has received funding from the European Union's Horizon research and innovation programme under grant agreement **No 101082624**