Implementation of use cases on top of OPENCOSS platform

D1.4

<table>
<thead>
<tr>
<th>Work Package:</th>
<th>WP1: Use Case Specification and Benchmark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dissemination level:</td>
<td>CO</td>
</tr>
<tr>
<td>Status:</td>
<td>Final</td>
</tr>
<tr>
<td>Date:</td>
<td>March 30, 2015</td>
</tr>
<tr>
<td>Responsible partner:</td>
<td>Huáscar Espinoza (TECNALIA)</td>
</tr>
<tr>
<td>Contact information:</td>
<td><a href="mailto:huascar.espinoza@tecnalia.com">huascar.espinoza@tecnalia.com</a></td>
</tr>
</tbody>
</table>
Contributors

<table>
<thead>
<tr>
<th>Names</th>
<th>Organisation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabien Belmonte, Laurent Pitot-de-la-Beaujardiere</td>
<td>ALSTOM Transport</td>
</tr>
<tr>
<td>Eric Verhuslt, Leonel Braga</td>
<td>ALTREONIC</td>
</tr>
<tr>
<td>Alberto Melzi</td>
<td>Centro Ricerche Fiat S.C.p.A.</td>
</tr>
<tr>
<td>Giorgio Tagliaferri, Vincenzo Manni, Andrea Critelli</td>
<td>RINA Services SpA</td>
</tr>
<tr>
<td>Jose Luis de la Vara, Sunil Nair</td>
<td>Simula Research Laboratory</td>
</tr>
<tr>
<td>Huáscar Espinoza, Xabier Larrucea, Idoya del Rio, Angel Lopez, Alejandra Ruiz, Ma. Carmen Palacios</td>
<td>TECNALIA</td>
</tr>
<tr>
<td>Sandrine Georges, Cedric Chevrel, Franck Aimé, Cyril Marchand</td>
<td>THALES Avionics</td>
</tr>
<tr>
<td>Tim Kelly, Oleg Lisagor, Katrina Atwood, Thomas Richardson</td>
<td>University of York</td>
</tr>
<tr>
<td>Mark van den Brand</td>
<td>TU/e</td>
</tr>
<tr>
<td>Eric Boccanfuso, Patrick Robert</td>
<td>Atego France</td>
</tr>
</tbody>
</table>

Document History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Remarks</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>V0.1</td>
<td>2014-11-15</td>
<td>Document creation and initial ToC</td>
<td></td>
</tr>
<tr>
<td>V0.2</td>
<td>2014-12-01</td>
<td>First draft on ALT contributions</td>
<td></td>
</tr>
<tr>
<td>V0.3</td>
<td>2014-12-19</td>
<td>TU/e contributions added, SIM contributions added, CRF contributions added, TAV contributions added</td>
<td></td>
</tr>
<tr>
<td>V0.4</td>
<td>2015-03-18</td>
<td>Avionics and Automotive case study documentation</td>
<td></td>
</tr>
<tr>
<td>V0.5</td>
<td>2015-03-19</td>
<td>Intro and background sections</td>
<td></td>
</tr>
<tr>
<td>V0.6</td>
<td>2015-03-24</td>
<td>Full version for review</td>
<td></td>
</tr>
<tr>
<td>V1.0</td>
<td>2015-03-30</td>
<td>Final version</td>
<td></td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

Abbreviations ...................................................................................................................... 7

Executive Summary .............................................................................................................. 8

1 Introduction ...................................................................................................................... 9
   1.1 Scope and Purpose ..................................................................................................... 9
   1.2 Relationship with other Deliverables .................................................................. 10
   1.3 Structure of the Document ..................................................................................... 10

2 Background ..................................................................................................................... 11
   2.1 OPENCOSS Prototyping Roadmap ..................................................................... 11
       2.1.1 Prototype 1 .................................................................................................... 12
       2.1.2 Prototype 2 .................................................................................................... 12
       2.1.3 Prototype 3 .................................................................................................... 13
   2.2 OPENCOSS Usage Scenarios per Application Domain ....................................... 15
   2.3 Main OPENCOSS Tool Concepts ........................................................................ 16
   2.4 Challenges for Implementing OPENCOSS Case Studies ..................................... 18
       2.4.1 Possibility of Evaluating some Aspects only in Real Projects ......................... 18
       2.4.2 Lack of Information about the Current Situation ............................................ 18
       2.4.3 Need for More Knowledge about Specific OPENCOSS Results ................... 19
       2.4.4 Confidentiality Issues .................................................................................... 19

3 Automotive Case Study ................................................................................................. 20
   3.1 Case Study Specification ....................................................................................... 20
   3.2 Case Study Source Data ......................................................................................... 20
   3.3 Case Study Implementation in OPENCOSS ............................................................ 25
       3.3.1 Modelling of the ISO 26262 reference framework ........................................ 25
       3.3.2 Creation of the Assurance Project ................................................................. 25
       3.3.3 Modeling of the Project Artefacts .................................................................. 29
       3.3.4 Compliance management ................................................................................ 31
       3.3.5 Demonstration of the Vocabulary and Compositional Argument Approaches in the Automotive Case Study .......................................................... 33
   3.4 General conclusion and improvement ..................................................................... 47

4 Avionics Case Study ........................................................................................................ 48
   4.1 Case Study Specification ....................................................................................... 48
   4.2 Case Study Source Data ......................................................................................... 48
       4.2.1 Railway Source Data ...................................................................................... 48
       4.2.2 Avionics Source Data ..................................................................................... 50
   4.3 Case Study Implementation in OPENCOSS ............................................................ 51
       4.3.1 Avionics and Railway Safety Standard Reference Framework Modeling ........ 51
       4.3.2 Equivalence Mapping Modeling ..................................................................... 57
       4.3.3 Company Project Modeling ........................................................................... 60
       4.3.4 Reuse of railways artefacts in avionics domain ................................................. 61
   4.4 General conclusion and improvement ..................................................................... 65

5 Railway Case Study ......................................................................................................... 66
   5.1 Case Study Specification ....................................................................................... 66
   5.2 Case Study Source Data ......................................................................................... 66
       5.2.1 CENELEC Standards ..................................................................................... 66
       5.2.2 Belgium GATC Assurance Project ................................................................. 68
       5.2.3 France GATC Assurance Project ................................................................... 73
   5.3 Case Study Implementation in OPENCOSS ............................................................ 74
5.3.1 Modelling of the EN standards ................................................................. 74
5.3.2 Specification of the Assurance Project (Belgium) ........................................ 77
5.3.3 Safety Case Modelling .............................................................................. 79
5.3.4 Compliance Management .......................................................................... 84
5.3.5 Cross-project reuse to create a new Assurance Project (France) .............. 86
5.3.6 Case Study Implementation from the external Process Tool "Process Director" ........................................................................................................ 88
5.1 General conclusion and improvement .......................................................... 95

6 Conclusions ..................................................................................................... 96
List of Figures

Figure 1. General Prototype Schedule................................................................. 11
Figure 2. Summary of the Roadmap for OPENCOSS Prototypes.......................... 14
Figure 3. Screenshot of the Eclipse editor of the OPENCOSS tool platform .......... 17
Figure 4. Screenshot of the Web application of the OPENCOSS tool platform ...... 18
Figure 5. Item definition index and assumptions list ........................................ 21
Figure 6. Block diagram of the electric parking system .................................... 21
Figure 7. Content of “Item Definition” and “HARA” sheets deployment .............. 22
Figure 8. Example of “HARA” outcomes sheet ................................................. 22
Figure 9. Example of safety requirements sheet until FSRs ............................... 23
Figure 10. SEooC development at “system level” ............................................. 23
Figure 11. Vehicle functional safety requirements ........................................... 24
Figure 12. ISO 26262 Part 3 Reference framework modelled in OPENCOSS platform.............................................................................................................. 25
Figure 13. Baseline from ISO 26262 ................................................................ 26
Figure 14. Assurance project structure on the repository (automotive use case example) ........................................................................................................... 26
Figure 15. Assurance project representation example ........................................ 27
Figure 16. Assurance project excerpt representation focused on HARA ............ 28
Figure 17. Assurance project evidence section .................................................. 29
Figure 18. Information stored about HARA evaluation ...................................... 29
Figure 19. Actual Item Definition document link .............................................. 30
Figure 20. Assurance project “Assets package” section ...................................... 30
Figure 21. Mapping setting .............................................................................. 31
Figure 22. Mapping window ............................................................................ 31
Figure 23. Link of the Item Definition artefact model to the corresponding node on the baseline ................................................................. 32
Figure 24. GAP analysis example .................................................................... 32
Figure 25. Compliance report example ............................................................ 33
Figure 26. Screenshot of the Vocabulary Model for ISO 26262, focussing on the general categories which are used to structure the vocabulary .................................................................................. 35
Figure 27. Screenshot of the Vocabulary Model for ISO26262, illustrating the modelling of terms ................................................................. 35
Figure 28. Modular Argument Structure .......................................................... 37
Figure 29. EPark Component Argument Module ............................................ 39
Figure 30. FSR_Argument Module .................................................................. 40
Figure 31. PowerSupply_Argument Module .................................................... 40
Figure 32. Contract Module to Resolve eParkComponent.G9 ......................... 41
Figure 33. VCU_Argument Module .................................................................. 41
Figure 34. GSM_Argument Module .................................................................. 42
Figure 35. Contract Module to Resolve eParkComponent.G6 ......................... 42
Figure 36. Contract Module to Resolve eParkComponent.G7 ......................... 43
Figure 37. Partial list of generic claim types in the Claim Types Editor .............. 44
Figure 38. Selecting Goals to be Resolved by Contract .................................... 45
Figure 39. Selecting Supporting Elements ....................................................... 46
Figure 40. Selecting a Justification for a Mapping by means of a Contract .......... 46
Figure 41 Avionics Life Cycle Data .................................................................. 50
Figure 42 Dual use of the case study ............................................................... 51
Figure 43 Safety Standards Documents framework ....................................... 52
Figure 44 Safety Standards Definitions analysis .............................................. 53
Figure 45 Safety Standards Strategy analysis .................................................. 54
Figure 46 Avionic Safety Standard based on control loop principle, double and reversed ................................................................................................. 54
Figure 47 : OPENCOSS Tool : Tree view of EN50128 model......................... 55
Figure 48 : OPENCOSS Tool : Tree view of DO178C model, focused on artefacts and their constraining requirements ................................................. 56
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>49</td>
<td>Step 1: Standard appropriation</td>
</tr>
<tr>
<td>50</td>
<td>Equivalence mapping approach</td>
</tr>
<tr>
<td>51</td>
<td>Step 2: Feature analysis and equivalent objective traceability =&gt; PxB standard</td>
</tr>
<tr>
<td>52</td>
<td>Completeness toward common performance based abstraction</td>
</tr>
<tr>
<td>53</td>
<td>Equivalence Mapping in Opencoss tool</td>
</tr>
<tr>
<td>54</td>
<td>Technical selection process</td>
</tr>
<tr>
<td>55</td>
<td>Mapping ENS0128 requirements to evidences</td>
</tr>
<tr>
<td>56</td>
<td>Identifying evidences compliance map</td>
</tr>
<tr>
<td>57</td>
<td>Building Compliance Map to DO178C using the Cross Domain Opencoss tool feature</td>
</tr>
<tr>
<td>58</td>
<td>Safety Case Structure according to EN 50129</td>
</tr>
<tr>
<td>59</td>
<td>Example of System-life cycle according to EN 50126</td>
</tr>
<tr>
<td>60</td>
<td>Safety Assurance Activities associated with V-Cycle Phases</td>
</tr>
<tr>
<td>61</td>
<td>Table of Content of the Safety Plan for a ERTMS Application project (Signalling System)</td>
</tr>
<tr>
<td>62</td>
<td>Excerpts of a Hazard Log for the GATC trainborne subsystem</td>
</tr>
<tr>
<td>63</td>
<td>Global Fault Tree Structure</td>
</tr>
<tr>
<td>64</td>
<td>PHA Example of Table</td>
</tr>
<tr>
<td>65</td>
<td>SSHA Example of Table</td>
</tr>
<tr>
<td>66</td>
<td>Safety Assessment Report content for the Dynamic Configuration of the EVC</td>
</tr>
<tr>
<td>67</td>
<td>Modeling of EN 50126 (Graphical view)</td>
</tr>
<tr>
<td>68</td>
<td>Example of Reference Requirements for EN 50126 (textual view)</td>
</tr>
<tr>
<td>69</td>
<td>Excerpts of the modeling of EN 50128 (Graphical View)</td>
</tr>
<tr>
<td>70</td>
<td>Excerpts of Recommendation tables of EN 50128 modeled in Opencoss</td>
</tr>
<tr>
<td>71</td>
<td>The Belgium GATC assurance project with the two Baseline models (EN 50126 and 50128)</td>
</tr>
<tr>
<td>72</td>
<td>Excerpt of the Artefact Models for the Belgium GATC assurance project</td>
</tr>
<tr>
<td>73</td>
<td>Global structure of ERTMS Safety Case models (Opencoss on top of the Figure)</td>
</tr>
<tr>
<td>74</td>
<td>Safety Case model: GATC trainborne subsystem is adequately safe</td>
</tr>
<tr>
<td>75</td>
<td>Safety Case model: GATC behaviour in absence of failures is safe &amp; GATC faults are sufficiently mitigated</td>
</tr>
<tr>
<td>76</td>
<td>Safety Case model: The Hazards are Sufficiently Mitigated</td>
</tr>
<tr>
<td>77</td>
<td>Safety Case model: Defence against systematic failures in GATC is adequate</td>
</tr>
<tr>
<td>78</td>
<td>Compliance maps for GATC project artefacts required by the EN 50126 standard</td>
</tr>
<tr>
<td>79</td>
<td>Compliance report for GATC project artefacts required by the EN 50126 standard</td>
</tr>
<tr>
<td>80</td>
<td>Gap analysis report for GATC project artefacts required by the EN 50126 standard</td>
</tr>
<tr>
<td>81</td>
<td>Metrics related to argumentation in the GATC assurance project</td>
</tr>
<tr>
<td>82</td>
<td>Cross-project reuse from Belgium GATC to France GATC assurance project</td>
</tr>
<tr>
<td>83</td>
<td>Compliance map after reusing GACT Belgium project into GATC France project</td>
</tr>
<tr>
<td>84</td>
<td>EN-50126 Library Structure in Process Director</td>
</tr>
<tr>
<td>85</td>
<td>Example of EN-50126 reference activity with its CCL attributes</td>
</tr>
<tr>
<td>86</td>
<td>Extract of EN-50126 BPMMN Structure in Process Director</td>
</tr>
<tr>
<td>87</td>
<td>CCL conformance result before importing EN-50126 into Opencoss</td>
</tr>
<tr>
<td>88</td>
<td>Extract of EN-50128 Library Structure in Process Director</td>
</tr>
<tr>
<td>89</td>
<td>Example of EN-50128 reference activity with its CCL attributes</td>
</tr>
<tr>
<td>90</td>
<td>Top of EN-50128 BPMMN Structure in Process Director</td>
</tr>
<tr>
<td>91</td>
<td>CCL conformance result before importing EN-50128 into Opencoss</td>
</tr>
<tr>
<td>92</td>
<td>GATC France Library Structure in Process Director</td>
</tr>
<tr>
<td>93</td>
<td>CCL conformance result before importing GATC France into Opencoss</td>
</tr>
</tbody>
</table>
Abbreviations

ARTEMIS  Advanced Research & Technology for Embedded Intelligence and Systems
CCL    Common Certification Language
CENELEC Comité Européen de Normalisation Electrotechnique (European Committee for Electrotechnical Standardization)
CESAR Cost-efficient methods and processes for safety relevant embedded systems
DACS Data & Analysis Center for Software
DoW  OPENCOSS Description of Work
DX.Y OPENCOSS deliverable X.Y
ERTMS European Railway Traffic Management System
ETCS European Train Control System
EVC European Vital Computer
EVM Earned Value Management
FMEA Failure Mode and Effects Analysis
FTA Fault Tree Analysis
GATC Generic Automatic Train Control trainborne
GQM Goal/Question/Metric
IEEE Institute of Electrical and Electronics Engineers
IMA Integrated Modular Avionics
ISA Independent Safety Assessor
ISO International Organization for Standardization
NASA National Aeronautics and Space Administration
OMG Object Management Group
QA Quality Attribute
RECOMP Reduced Certification Costs Using Trusted Multi-core Platforms
ROI Return On Investment
RQ Research Question
R&D Research and Development
SAA Safety Assurance Asset
SEooC Safety Element out of Context
SLR Systematic Literature Review
S&T Scientific and Technical
TX.Y OPENCOSS task X.Y
V&V Verification and Validation
WP OPENCOSS work package
Executive Summary

This document (D1.4) aims to specify and implement case studies in the automotive, avionics, and railway domains as reference proofs-of-concept of OPENCOSS results. The overall goal of D1.4 is to develop the case studies over the OPENCOSS solution and following the OPENCOSS conceptual framework. The case studies are being incrementally developed by using intermediate tool prototypes and the final version of the integrated OPENCOSS tool platform.

The case studies (described in D1.2) are (1) an ePARK system for an electric vehicle in the automotive domain, (2) reuse of a railway execution platform in the avionics domain, and (3) the certification of a signalling system in the railway domain. This report summarises the set of data used to implement the case studies, as well as the results of the procedures followed to use OPENCOSS in each of the case studies. Finally, the report describes some limitations or assumptions in using OPENCOSS tools for the final prototype version. The results of this deliverable serve as basis for the next WP1 task (T1.5 – OPENCOSS Benchmarking).

The case studies are being incrementally developed by using intermediate tool prototypes and the final version of the integrated OPENCOSS tool platform. This deliverable contains a summary of case study implementation from the final (third) OPENCOSS tool platform prototype.
1 Introduction

1.1 Scope and Purpose

Safety assurance and certification are amongst the most expensive and time-consuming tasks in the development of safety-critical embedded systems. European innovation and productivity in this market is curtailed by the lack of affordable (re)certification approaches. Major problems arise when evolutions to a system entail reconstruction of the entire body of certification arguments and evidence. Further, market trends strongly suggest that many future embedded systems will be comprised of heterogeneous, dynamic coalitions of systems of systems. As such, they will have to be built and assessed according to numerous standards and regulations. Current certification practices will be prohibitively costly to apply to this kind of embedded systems.

The OPENCOSS project aims to devise a common certification framework that spans different vertical markets for railway, avionics and automotive industries, and to establish an open-source safety certification infrastructure (hereafter referred to as OPENCOSS tool platform). The infrastructure is being realised as a tightly integrated solution, supporting interoperability with existing development and assurance tools. The ultimate goal of the project is to bring about substantial reductions in recurring safety certification costs, and at the same time increase product safety through the introduction of more systematic certification practices. Both will boost innovation and system upgrades considerably.

WP1 is concerned with the specification and implementation of case studies as reference proofs-of-concept of OPENCOSS results. These results are the conceptual framework and the tool platform to be specified and developed in the technical WPs (WPs 2, 3, 4, 5, 6, and 7). The conceptual framework will mainly correspond to the CCL and the compositional certification conceptual framework specified in WP4 and WP5, respectively. The tool platform will mainly correspond to the safety certification management infrastructure developed in WP6 and WP7, although other implementation aspects will be addressed in the rest of technical WPs. Once OPENCOSS results were available, they have been benchmarked with the aid of three case studies, each focusing on one of the application domains addressed in the project.

This document (D1.4) aims to specify and implement case studies in the automotive, avionics, and railway domains as reference proofs-of-concept of OPENCOSS results. The overall goal of D1.4 is to develop the case studies over the OPENCOSS solution and following the OPENCOSS conceptual framework.

The case studies (described in D1.2) are (1) an ePARK system for an electric vehicle in the automotive domain, (2) reuse of a railway execution platform in the avionics domain, and (3) the certification of a signalling system in the railway domain. This report summarises the set of data used to implement the case studies, as well as the results of the procedures followed to use OPENCOSS in each of the case studies. Finally, the report describes some limitations or assumptions in using OPENCOSS tools for the final prototype version. The results of this deliverable serve as basis for the next WP1 task (T1.5 – OPENCOSS Benchmarking).

The case studies were incrementally developed by using intermediate tool prototypes and the final version of the integrated OPENCOSS tool platform. This deliverable contains a summary of case study implementation from the final (third) OPENCOSS tool platform prototype.
1.2 Relationship with other Deliverables

D1.4 is related to other OPENCOSS deliverables, which have served as input, with which consistency must be kept, or that will use its results. These deliverables, and the relationship of D1.4 with them, are the following ones:

- D1.1 (Constraints of the certification process) and D2.1 (Business cases and user needs) present general information about the safety assurance and certification processes in the automotive, avionics, and railway domains.
- D1.2 (Use cases descriptions and business impact) describes in depth the case studies with which OPENCOSS results will be evaluated.
- D3.1 (Analysis of safety certification data of industrial use cases) describes current safety assurance and certification practices in the automotive, avionics, and railway domain, and provides data of the case study for each domain.
- Deliverable D2.4 provides a set of generic OPENCOSS usage scenarios and domain-specific usage scenarios modelled as BPMN diagrams, which are the basis for the case study implementation.

1.3 Structure of the Document

The rest of the deliverable is structured as follows. Section 2 discusses the background information and work necessary for understanding the case study implementation and the underlying concepts. Sections 3, 4 and 5 show how the OPENCOSS platform has been used in the automotive, avionics, and railway case studies, respectively. Section 7 presents the conclusions of the deliverable.
2 Background

This section presents some background information that might be necessary to understand D1.4.

First, we present the OPENCOSS case study implementation roadmap. Next, a summary of the Usage Scenarios per application domain is described. The OPENCOSS tooling main concepts are also discussed. Finally, some challenges for implementing the OPENCOSS Case Studies are analysed.

2.1 OPENCOSS Prototyping Roadmap

The OPENCOSS project targets high-risk but high pay-off objectives. In order to mitigate these risks, it is of high value to follow an incremental approach by developing rapid and early prototypes. As a general strategy, the OPENCOSS consortium decided to follow an incremental approach for research and development, unlike the DoW approach, which did not take into account the evolution of the different solutions. The benefits of following a prototyping approach are:

- Better assessment of ideas by focusing on a few aspects of the solution
- Ability to change critical decisions by using practical and industrial feedback (case studies)
- Ability to assess feasibility of solutions
- More advantage on the availability of the associate domain knowledge (pointing out details that might otherwise be less visible)
- Better planning for project evolution and benchmarking

The OPENCOSS project released three prototype iterations. Figure 1 shows the schedule of the three prototype iterations since the tool implementation started. We can see that there is some overlapping in time, which is due to the fact that some work is already running in the next prototype when tool evaluation and refinement tasks are running.

When we refer to “prototypes” in OPENCOSS, we mean three prototyping dimensions:
1. **Conceptual/Research Development**: development of solutions from a conceptual perspective.
2. **Tool Implementation**: implementation of tools implementing conceptual solutions.
3. **Case Study Implementation**: deployment of industrial case studies using the conceptual and tooling solutions.

The overall roadmap for the three prototype iterations can be summarised as follows:

### 2.1.1 Prototype 1

**Goal**: The main goal of this prototype iteration is to validate the CCL Metamodel (first stable version) with regard to its power of expression in order to capture some fundamental information from the three industrial domains (automotive, avionics, railway) to enable compliance management, argument specification/modularisation, evidence characterisation and process assurance.

**Preconditions**: Common understanding of the business scenarios (for the three industrial domains), requirements for the OPENCOSS platform, existing solutions (information models, tools, methods, etc.), and a preliminary solution outline in terms of CCL metamodel as well as OPENCOSS tool architecture.

**Post Conditions**: A basic prototype infrastructure for editing and navigating assurance-related information, and recommendations to tune CCL in order to enable the next prototyping goal: assurance assets reuse.

**Conceptual/Research Development**: first CCL draft containing features for standards modelling, evidence characterization, process modelling, argumentation modelling, and compliance specification.

**Tool Implementation**: The Prototype 1 version of the OPENCOSS tools focused on CCL implementation. In particular:
- Graphical editors to represent information such as Standards and Company-specific Processes (Both also referred to as Reference Frameworks) and Argumentation.

**Case Study Implementation**: Data from industrial partners covered different aspects of the assurance and certification process and in some cases have different levels of granularity, detail and completeness. Prototype 1 focused on evaluating the following aspects:
- Availability of industrial data to perform a meaningful excerpt of the case studies.
- Ability to extract the required information from case studies data so as to model them to achieve OPENCOSS goals (guidance, reuse, process automation, etc.).

**Execution Period**: Beginning of the project – November 2013

### 2.1.2 Prototype 2

**Goal**: The main goal of this prototype iteration is to consolidate the CCL metamodel and provide a basic approach for cross-domain reuse, specification of contract-based compositional assurance, traceability and impact analysis as well as a preliminary proof of concept of the integration with process management tools.

**Preconditions**: A basic prototype infrastructure for editing and navigating assurance-related information with a stable version of CCL.
Post Conditions: A complete OPENCOSS tool infrastructure covering most of the planned functionalities.

Conceptual/Research Development: In Prototype 2, CCL has been completed to cover cross-domain reuse by including the “equivalence map” concept, together with the “compliance post-conditions” attached to equivalence maps. Also, the compositional assurance specification approach now allows tool users to specify assumptions, guarantees and contracts when integrating components into larger systems. Finally, an “artefact traceability” approach has been defined in this iteration in order to keep tracking of changes done and impact resulting from artefact modifications.

Tool Implementation: The Prototype 2 version of the OPENCOSS tools focused on integrating the whole set of tool data into a common database. In particular:

- Provide specific implementation for compliance management as Web services, traceability, impact analysis.
- Integrate existing Eclipse-based client modules and communication between them: database storage, CDO layer, web server, infrastructure for API server.

Case Study Implementation: Industrial case studies centered on specific modeling aspects:

- The avionics case study studied a basic approach for cross domain reuse, which is based on the Equivalence Mapping concept. As a first step, we modelled the equivalence between CEN 50128 and DO-178C standards at the level of “compliance requirements” and “artefacts”
- The automotive case study focused on modelling SEooC assumptions, which is the basis for the contract-based approach developed by OPENCOSS.
- The railway case study focused on modelling the safety case for a subsystem, using the OPENCOSS argumentation editors.


2.1.3 Prototype 3

Goal: The main goal of this prototype iteration is to complete the full OPENCOSS functionality for cross-domain reuse, and allow creating contracts for the compositional approach. Besides, the Web clients shall be completed to cover the compliance reports, metrics measurements and gap analysis. Finally, the connection with process management tools shall be demonstrated.

Preconditions: Data-centralized OPENCOSS tool architecture with feedback from case studies to refine cross-domain reuse and compositional assurance functionalities.

Post Conditions: Full OPENCOSS tools to be released to an open-source community.

Conceptual/Research Development: In Prototype 3, the CCL Vocabulary approach has been developed as well as the mechanisms to validate “contracts” in the context of the compositional assurance approach. Also, the cross-domain and cross-project approaches were refined to allow users take reuse decisions in a more assisted way. Additionally, the link with the Atego Process Director tool has been conceptually refined to get a more precise translation of concepts to the OPENCOSS platform.

Tool Implementation: The Prototype 3 version of the OPENCOSS tools focused on refining the tools functionality supported on case study proofs from Prototype 2. This includes:

- Cross-domain reuse approach enhanced by adding justification of mappings and compliance post-conditions/obligations when creating reuse links.
- Integration of contract based approach with the argumentation framework of OPENCOSS
- Tools for Vocabulary specification on arguments
- Improvement on support to artefact lifecycle management.
- Allow impact analysis engine working on server side and client side.
- Allow impact analysis engine triggering and results presentation after changing artefact on client side.
- Link of OPENCOSS with Medini Analyse tools (safety analyses tool)
- Evidence assessment tools developed in Eclipse
- Metrics measurement server to be used with Web clients.

**Case Study Implementation**: Industrial case studies centered on specific modeling aspects:

- The avionics case study extended the cross-domain reuse approach to cover the whole set of equivalence mappings between CEN 50128 and DO-178C standards, and used the concept of compliance post-conditions/obligations to complete the assistance for reuse decisions.
- The automotive case study focused on modelling the whole set of compositional concepts: assumptions, guarantees and contracts. The ePark subsystem was used as a SEooC integrated in a vehicle. This completes the compositional approach for assurance
- The railway case study focused on modelling the two assurance projects used to measure cross-country reuse efforts with OPENCOSS.

**Execution Period**: May 2014 – March 2015

![Figure 2. Summary of the Roadmap for OPENCOSS Prototypes](image)

Figure 2 synthetises the roadmap for the three OPENCOSS prototype iterations along with tool enhancement and opportunities, which were not planned in the initial OPENCOSS Description of Work (DoW). Two enhancement opportunities are:

- OSLC-based integration: creation of an OSLC adaptor for integrating OPENCOSS with Medini Analyse (safety analyses tool for the automotive domain).
• Tool security, reliability and qualification: we were unable to develop these functionalities. They should be studied in future projects or during the integration of OPENCOSS into an open-source community.

2.2 OPENCOSS Usage Scenarios per Application Domain

Usage Scenarios describe user interactions with a system to capture the context and goals of the user. The focus of a Usage Scenario is on the User, and specifically on the tasks they need to accomplish toward achieving high-level goals.

The approach to specify usage scenarios is based on the following principles:

(a) The description of usage scenarios is centred on the OPENCOSS platform “user” perspective (how users will interact with the OPENCOSS platform), in the context of typical business cases (which safety assurance and certification activities are involved).

(b) We provide two detail levels for the description of usage scenarios. A first high-level description offers a link between business process and OPENCOSS tool usage activities. The second lower level description identifies some OPENCOSS use cases (from Deliverable D2.3) and offers a set of related scenarios illustrating how users interact with the OPENCOSS tool platform.

The business cases are organized according to three drivers: unification, innovation, and reuse. Unification contributes to the common understanding between stakeholders. Innovation leads to faster and better assessment processes and results. Reuse addresses assessment data of already approved systems or system components. The business cases have been described based on the used engineering, assurance and certification activities and the involved stakeholders.

The domain specific usage scenarios, from avionics, railway, and automotive, have been developed based on the following inputs:

• (Business) Stakeholders and OPENCOSS Users (Tool Users).
• Goals per industrial case study (e.g. Cross-domain reuse).
• List of activities at “business” level.
• General Use Scenarios.

The Avionics Use Case has the following business cases:

1. Unification and update of the process and requirements.
2. Reuse of the assessment data of already approved components.
3. Innovation based on cross-over effects between application domains and applying theoretical knowledge in practice.

The Automotive Case Study has the following business cases:

1. Maximize safety and minimise safety-related business risks and costs while the complexity of car systems and components increase and costs for safety assessments should not increase.
2. Introduce a more stable, reliable, trustworthy valuation system that has a well-defined relation with safety.
3. Decrease number of claims by using the ISO 26262 standard.

The Railway Case Study has the following business cases:
1. Reuse of the safety case of a previous version or similar product, offers a major time and effort reduction on safety assessments, especially when the re-assessment is done by a different assessors.
2. Use of the OPENCOSS Platform will mean that recertification is facilitated when manufacturers produce identical products for another country.
3. Use of the OPENCOSS Platform could provide the developer with a better view of the relationships between requirements and system specification and implementation.

For further details on OPENCOSS usage scenarios, please see deliverable D2.4.

2.3 Main OPENCOSS Tool Concepts

As described in D2.3 (OPENCOSS Platform Architecture), the OPENCOSS tool platform has been developed around five main functional groups:

- Prescriptive Knowledge Management, related to the management (edition, search, transfer..) of standards’ information as well as any other information derived from them, such as interpretations about intents and mapping between standards.
- Assurance Project Lifecycle Management, related to the creation of safety assurance projects locally in OPENCOSS and any project baseline information.
- Safety Argumentation Management, related to the argumentation information of an assurance project, including modular and compositional argumentation.
- Evidence Management, related to the lifecycle of the artefacts used as evidence, including their traceability and impact analysis.
- Process Assurance Management, related to the specification, storage, and validation of safety assurance processes in connection with engineering processes, and to compliance information related to safety standards.

These functional groups have been refined in WP4-7 requirements and design deliverables (e.g., D6.2 - Detailed requirements for evidence management of the OPENCOSS platform, and D6.3 - Specification of the evidence management service infrastructure), and materialised in the corresponding implementation tasks (e.g., T6.4 - Implementation of the evidence management service infrastructure). The CCL also supports the information needs of the functional groups, and thus it has been structured in several metamodels targeted at specific needs (e.g., process metamodel; see D4.4). This has led to a CCL-based implementation of the functional groups, and resulted in an OPENCOSS tool platform whose use is based on the creation and maintenance of models for:

- Reference frameworks, which capture the high-level concepts and relationships against which the safety aspects of a given system are developed and assessed.
- Assurance projects, which define the assets produced during the development, assessment and justification of a safety-critical system, including those associated with justifying the safety of the system and – in the regulated domains – seeking regulatory approval for its entry into service.
- Baselines, which contain the parts of a reference framework with which compliance has to be shown in an assurance project, and the possible refinement or extension of the framework for project-specific purposes.
- Argumentation, which describes how features of a generic assurance argument are linked together, and how evidence can be used to support the reasoning presented in the argument. Tooling has also been provided to support the construction of modular, compositional arguments for large-scale assurance projects, including the automatic generation of contract modules linking argumentation modules dealing with different aspects of the system.
- Evidence, which defines the metadata captured about the artefacts of an assurance project
- Processes, which capture the high-level concepts relating to the processes to be performed as part of an assurance project and the artefacts that are produced by them.
• Vocabulary models which capture the structure of domain-specific vocabularies and structured expressions for use in tailored argument creation. Models are also used for specifying properties (e.g., of artefacts) and mappings. The mappings can correspond to equivalence maps (between reference frameworks, or between a baseline and a reference framework) and compliance maps (between an assurance project’s assets and its baseline).

A general usage scenario of the OPENCOSS tool platform would be as follows:

1. A company would model one or several reference frameworks for the standards to demonstrate compliance with.
2. For a specific assurance project, the company would create a baseline based on some reference framework.
3. The company would then collect evidence and process information, and model the structured argumentation for the project. Evidence information would be associated to process and argumentation information (e.g., activities input and output, and information supporting a safety claim, respectively).
4. Finally, the company would have to specify how the evidence, process, and argumentation information show the compliance with the project’s baseline.

Further, more detailed usage scenarios can be found in D2.4 (OPENCOSS Usage Scenarios).

From a more technological perspective, the OPENCOSS tool platform has been realised in the form of:

• Eclipse-based editors (Figure 3), for creating and maintaining reference framework, assurance project, baseline, safety argumentation, evidence, and process models.
• Web application (Figure 4), which synthesize and summarise compliance information by mean of different reports (e.g., gap analysis report), and can also be used for consulting the baseline, evidence, process, and argumentation information of an assurance project.

Both parts use a common server and data storage. The platform also allows the import of information from external tools for evidence and process management by means of an API.

![Figure 3. Screenshot of the Eclipse editor of the OPENCOSS tool platform](image-url)
### Challenges for Implementing OPENCOSS Case Studies

This section discusses the main challenges that we have found for implementing the case studies.

#### 2.4.1 Possibility of Evaluating some Aspects only in Real Projects

Among other objectives, WP1 in general and the enactment of the evaluation framework in particular aim to show the benefits from using OPENCOSS results in current practice on safety assurance and certification. Therefore, the aim was to try to measure several metrics (see D1.3) in the case studies of each application domain.

In practice, the only way to really compare the situation before and after the availability of the OPENCOSS platform, would be to execute the same project twice. This is most often not economical and has methodological issues as well. For example the same team cannot be used as it would bias the second execution of the project. Hence, the most obvious method would be for a given organisation that has sufficient historical metrics, to compare how subsequent projects are executed and deliver after the OPENCOSS is introduced and used. An example of this was witnessed during the on-site survey whereby a development team switched from a traditional waterfall approach to an agile process. In this case the integration time and the residual errors, as significant metrics, were comparable and significantly improved.

Another aspect that compounds the comparison is that the reuse of components and assurance artefacts is only measureable over successive projects. The first project is likely not to have much benefit as the work has to be done once, but subsequent projects can benefit from it.

As a general conclusion, we can say that the evaluation of the benefits of using the OPENCOSS platform will mainly happen in the future in the assumption that metrics have been collected before its introduction.

#### 2.4.2 Lack of Information about the Current Situation

Recapitulating the results documented in D1.3, we confirm that the initial information gathered in the OPENCOSS project to analyse the current practices for safety assurance and certification (using desk
research and surveys) has highlighted that the task is perhaps more complex than anticipated. This is a result of the lack of details and of the lack of a common understanding for some aspects. The most important challenges found are as follows:

1. Safety standards against which certification is applied are complex and translated differently in each domain and organisation.
2. Many projects that relate to safety certification have a long lifecycle in combination with incremental modifications. It is not trivial to extract cost and resource data that can be allocated to a specific project only. In many cases this data is not recorded in a systematic way.
3. Practices are very diverse, ranging from paper driven top-down processes to lean and agile software engineering. Hence comparison is difficult.
4. In the automotive domain, certification is not really an issue yet. For the pre-cursor (conformity assessment), no systematic records are kept as the data is spread over the TIER-1-2-3 supply chain.
5. In the ideal case, the same project should be executed with and without OPENCOSS support and the resource and time consumption compared. In line with the second observation, this is not a reachable goal and can be prohibitive in cost.

Therefore it was proposed to carry out point comparisons. The reasons listed above as well as the fact that the OPENCOSS platform is not yet in a stable state, the only action possible is to verify that the OPENCOSS platform is able to meets its initially formulated requirements. We refer to D.3.3. for details.

2.4.3 Need for More Knowledge about Specific OPENCOSS Results

From a general perspective, the evaluation framework aimed to analyse the achievement of OPENCOSS goals (G1, G2, and G3) based on the new possibilities that OPENCOSS results enable for safety assurance and certification. Consequently, the framework directly depends on the conceptual and technological solutions that the project provides.

During the development, the OPENCOSS platform has evolved through several prototypes. At the time of writing acceptance test have been conducted (see D.3.3) using version 2 of the prototype. As the test indicate, several issues were found, including implementation issues. Hence, a full validation is dependent on the availability of prototype version 3. This hampers also the evaluation of the effectiveness of the OPENCOSS platform as it requires that use case data be entered and exploited.

2.4.4 Confidentiality Issues

Another issue encountered was that in many cases a complete data set is not available for confidentiality and competitive pressure reasons. As mitigation measures, the following action lines were agreed upon:

1. The industrial partners sanitise the case study data for approval.
2. The scope of evaluation and of V&V were initially narrowed to the domains of software verification and executing the hazard and risk analysis.
3. The industrial partners use the internally available data and provide the consortium with the measured or assessed improvement figures. This way no sensitive data had to be communicated.
3 Automotive Case Study

This section presents an implementation report of the Automotive Case Study as it was deployed to benchmark OPENCOSS approach and tools.

3.1 Case Study Specification

The OPENCOSS deliverable D1.2 describes the Automotive Case Study. It consists of a parking system for electric vehicles, used for blocking the wheels when the car is stopped in a parking area. The aim of the device is to avoid any unwanted movement of the vehicle during parking, due to the fact that an electric powertrain (electric motor with a reduction gear) is free to rotate even when not powered. This type of system involves electric/electronic, logical and mechanical aspects and is developed as an innovative concept available for different models of a vehicle family.

The design of this device, aimed to be reused for different electric vehicles, is conceived as a “Safety Element out of Context” or SEooC (ISO 26262-10). This means that it is expected to reuse the same system for several applications. For achieving this from the automotive conformity assessment point of view, it is necessary to apply the ISO 26262 standard to the design of the system and to check it with respect to the design of the target vehicle, also developed according to the ISO 26262 standard.

In other words, it is necessary to guarantee the compliance to the standard at the beginning of the system concept definition and then, during the actual application, to check if the compliance is still valid. The necessary compliance requirements at the vehicle level from the target application must be available.

3.2 Case Study Source Data

The set of data used for the implementation of the automotive case study are constituted by the ensemble of the information related to the functional description of the system under development and to its safety requirements derived during the application of the ISO 26262 (Road vehicles — Functional safety) standard process.

The standard itself constitutes the reference framework to which the system deployment refers from the point of view of the functional safety.

All the information is generally collected into Word/Excel documents.

The first document is constituted by the “Item definition” as requested by the standard. The primary scope of this document is to describe the system, by collecting all the information from the functional point of view with reference to the final user(s) (e.g. driver, passenger, pedestrian, repairer, services ...). This approach is helpful to evaluate in the correct way, in the next phase of the analysis (Hazard Analysis and Risk Assessment - HARA), the safety relevance associated to the system.

The content of the system description outlines its functional requirements, its dependencies on and interaction with the environment (operating scenarios) and other systems (preliminary architecture). In the case of the development of a generic system not immediately finalized to a target vehicle but conceptually designed for use in several application (e.g. various classes or types of vehicles), that ISO 26262 defines as SEooC (Safety Element out of Context), some assumptions have to be made about the
intended future use. These assumptions can be collected into the “Item Definition” word document or referred to a separate excel table.

Figure 5 shows the index of the item definition template is reported, together with a separate table of assumptions for the electric parking system intended for the current automotive use case in OPENCOSS application:

![Figure 5. Item definition index and assumptions list](image)

Figure 6 represents the block diagram of the preliminary architecture of the system inserted into the Item definition document for the electric parking system:

![Figure 6. Block diagram of the electric parking system.](image)

The second document is the report issued performing the Hazard Analysis and Risk Assessment (HARA). It is structured with an Excel table organized mainly in three sheets, which receives in input the information from the “Item definition”.

In Figure 7 is represented the schema of the relationships between the “Item definition” and the “HARA” sheets:
Implementation of use cases on top of OPENCOSS platform

D1.4

Figure 7. Content of “Item Definition” and “HARA” sheets deployment.

From “HARA” outcomes the Safety Goals (SG) are also listed: they are the top level safety requirements.

These last, anyway, constitute also the first entry of the general safety requirements chain that contains the Functional Safety Requirements (FSR) derived from the Safety Goals and the Technical Safety Requirements (TSR) derived from the FSRs.

The safety requirements chain is collected into an Excel table, containing also the allocation of the FSRs to the elements of the preliminary architecture and their ASILs derived from the SGs; also the TSRs have their respective ASIL assignment from the corresponding FSRs and their allocation to a more detailed system architecture consequent to TSRs definition.

Excerpts of the outcomes sheet from the “HARA” excel report is showed in Figure 8:

![Figure 8. Example of “HARA” outcomes sheet.](image_url)
In Figure 9 an excerpt of the safety requirements table is represented until FSRs:

![Safety Requirements Table](image)

Figure 9. Example of safety requirements sheet until FSRs.

The use case of the electric parking system is related to a “system level” deployment where the data set of the requirements is only available at FSRs level, according to the schema in Figure 10 from ISO 26262, and the FSRs are intended as “assumed” according to the assumptions from the “Item Definition”.

![System Level Diagram](image)

Figure 10. SEooC development at “system level”.
At this level, the functional safety requirements from the target vehicle are the last information available as listed in the Excel table of Figure 11.

<table>
<thead>
<tr>
<th>ID</th>
<th>Functional Safety Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSR1</td>
<td>VCU internal failure(s) [SW error(s), Electrical fault(s)] shall be diagnosed and VCU shall set an error status flag and transmit it on CAN network.</td>
</tr>
<tr>
<td>FSR2</td>
<td>VCU shall diagnose the failures on Electric Parking System button switches and related wired connections.</td>
</tr>
<tr>
<td>FSR3</td>
<td>GSM internal failure(s) [SW error(s), Electrical fault(s)] shall be diagnosed and GSM shall set an error status flag and transmit it to VCU via CAN network.</td>
</tr>
<tr>
<td>FSR4</td>
<td>GSM shall diagnose the failures on park pawl motor actuation signals.</td>
</tr>
<tr>
<td>FSR5</td>
<td>GSM shall diagnose the park pawl motor connections.</td>
</tr>
<tr>
<td>FSR6</td>
<td>GSM shut down shall be detected by VCU, checking the loss of communication.</td>
</tr>
<tr>
<td>FSR7</td>
<td>In presence of any fault, GSM shall set an error status flag and transmit it to VCU via CAN network.</td>
</tr>
<tr>
<td>FSR8</td>
<td>An engagement command affected by an error shall be cancelled, maintaining the park pawl disengaged.</td>
</tr>
<tr>
<td>FSR9</td>
<td>An incorrect engagement actuation shall be recovered by a new request of park pawl disengagement.</td>
</tr>
<tr>
<td>FSR10</td>
<td>If a parking request evaluation error is detected, the VCU avoids ePRND logic entering the parking mode through P button input.</td>
</tr>
<tr>
<td>FSR11</td>
<td>If a loss of communication between GSM and VCU occurs, GSM shall reach the last validated parking pawl acquisition target (with reduced motor speed), if any, or maintain current parking pawl state.</td>
</tr>
<tr>
<td>FSR12</td>
<td>The parking button LED have to be turned on/off only when the parking pawl is engaged/disengaged.</td>
</tr>
<tr>
<td>FSR13</td>
<td>If a VCU power supply fault occurs, the HMI turns on a warning light on dashboard and display to the driver warning or alarm message.</td>
</tr>
<tr>
<td>FSR14</td>
<td>VCU shall diagnose the failures on brake pedal switches and related wired connections.</td>
</tr>
<tr>
<td>FSR15</td>
<td>GSM shall send to VCU an electric parking system motor fault flag.</td>
</tr>
<tr>
<td>FSR16</td>
<td>A disengagement command affected by an error shall be cancelled, maintaining the parking pawl engaged.</td>
</tr>
<tr>
<td>FSR17</td>
<td>An incorrect disengagement actuation shall be recovered by a new request of parking pawl engagement.</td>
</tr>
<tr>
<td>FSR18</td>
<td>If a parking pawl motor error is detected during VCU to GSM handshake, the VCU shall invalid the handshake process and the GSM let park pawl to stay in its current position (if current position is park pawl engaged or unknown disable traction).</td>
</tr>
<tr>
<td>FSR19</td>
<td>Where the driver opens the door or unfolds seat belt an acoustic alarm and a message on the display shall alert the driver to insert the parking pawl before leaving the vehicle.</td>
</tr>
<tr>
<td>FSR20</td>
<td>Put a label in the cooldown, visible to the driver that warns him to block the vehicle before leaving it.</td>
</tr>
<tr>
<td>FSR21</td>
<td>Put a label on the charge port lid that warns the driver to block the vehicle before leaving it in charging.</td>
</tr>
<tr>
<td>FSR22</td>
<td>VCU shall diagnose the faults on electric parking system sensors (e.g., short circuit to power supply or ground).</td>
</tr>
<tr>
<td>FSR23</td>
<td>The system reaches the recovery state on handshake engage.</td>
</tr>
</tbody>
</table>

Figure 11. Vehicle functional safety requirements.

The matching of FSRs of the vehicle with the FSRs assumed for the electric parking system assures the validity of the assumptions and allows the integration of the SEooC into the target vehicle, for which the technical safety requirements of the electric parking system will be integrated in its technical safety requirements at the product development level.

The above described documentation constitutes the evidences for the project deployment implemented into the OPENCOSS platform for the electric parking system use case.

The evidences are supported by argumentations that are part of the content of the reports constituting the work products as required by the standard.
3.3 Case Study Implementation in OPENCOSS

3.3.1 Modelling of the ISO 26262 reference framework

The first step of implementation is the modelling of the ISO 26262 standard as the reference framework in the OPENCOSS platform. The possible future evolutions of the standard can be well taken into account, because the platform assures the necessary graphical and textual editors for making updated every reference framework.

The functional safety automotive standard is constituted by several parts reciprocally implicating. The electric parking system use case as SEooC is mainly limited to the Part 3 of the standard. The actual work anyway, for completeness involved the modelling of ISO 26262 Part 3 with the necessary implication from other Parts as 2 and 8 and also the Part 4 skeleton has been interested. The scope was to have a reference framework to which apply the tailoring for a baseline generation.

In Figure 12 an excerpt of the reference framework modelled is showed:

Figure 12. ISO 26262 Part 3 Reference framework modelled in OPENCOSS platform.

3.3.2 Creation of the Assurance Project

The baseline generation is the second step of the activity and is executed by tailoring the reference framework through a dedicated wizard menu in to the platform that allows selecting the desired parts of the reference framework according to the expected development. The baseline can be updated at any time according with the possible modification of the envisaged deployment.
The wizard window menu is showed in Figure 13 with the resulting baseline, which is again an articulated graphical and textual representation of a subset of the modelled standard, resulting in a tailoring of it for the application of the use case.

![Figure 13. Baseline from ISO 26262.](image)

Together with the baseline the entire structure of the assurance project is automatically generated containing (see Figure 14):

- The “Argumentation” folder with graphical and textual architecture description of the project as generated according to the baseline choice
- The “Assurance_project” folder containing the resulting baseline (graphical, as in Figure 9, and textual) and the assurance project structure that can be loaded with the modelled evidences (“artefact models”) into a specific “Asset package” section.
- The “Evidences” folder containing the modelled evidences for the above point.
- A “Processes” folder for the optional integration of external tools.

![Figure 14. Assurance project structure on the repository (automotive use case example).](image)
The assurance project is represented in a GSN format, putting in evidence in a tree the claims and the evidences supporting them.

The Figure 15 shows an example of this kind of tree, with the evidence applied to the bottom of the graphical representations:

![Assurance project diagram](image)

Figure 15. Assurance project representation example.
In the following picture an excerpt from the previous figure of the assurance project diagram has been showed focusing on the HARA part:

Figure 16. Assurance project excerpt representation focused on HARA.
3.3.3 Modeling of the Project Artefacts

The tree of the electric parking system is solved by the evidences represented by the content of the documents described in the previous section: “Item definition” (containing the SEooC assumptions), “HARA” report, List of Safety Goals and of the Functional Safety Requirements, together with the list of vehicle Functional Safety Requirements. Into the platform, these elements are modelled by the artefacts created on the “Evidence” folder, as above explained (see Figure 9.b), and represented in the following Figure 17:

![Figure 17. Assurance project evidence section.](image)

For the HARA, also the internal evaluation that was done has been stored.

![Figure 18. Information stored about HARA evaluation.](image)
The evidences artefacts (e.g. Item Definition as in the following picture) are linked with the resource of the documents which are stored on an SVN repository and so the evidences are linked with the actual documents. It is also possible to get the information about whom and when the latest modifications have been made to the document and open it.

![Figure 19. Actual Item Definition document link.](image)

Then the evidences artefacts are internally linked to the assurance project structure by the specific section “Assets package” of the “Assurance_project” folder as showed in the following Figure:

![Figure 20. Assurance project “Assets package” section.](image)
3.3.4 Compliance management

Finally, selecting the "baseline" element of the "Assurance_project" folder, a window appears with a "Mapping set" button (Figure 21):

![Figure 21. Mapping setting.](image)

Selecting this button, the compliance mapping menu is showed as in the following Figure 22, on which the available evidences (as "artefact models") can be linked to the nodes of the baseline previously generated.

![Figure 22. Mapping window.](image)
The following Figure 23 shows the details of the link of the “item definition” to the corresponding node on the baseline.

![Figure 23. Link of the Item Definition artefact model to the corresponding node on the baseline.](image)

In this way a link is created from the evidences and the baseline. This link close a loop where the baseline (what to do) is associated to the evidences loaded into the assurance project (what done), representing the “solutions” (GSN) of the activity tree.

Additionally, on the web the following information on the GAP analysis can be obtained, in order to see how many artefacts it is still needed to comply with:

![Figure 24. GAP analysis example.](image)
And the compliance report for each artefact can be also obtained:

![Compliance report example](image)

**Figure 25. Compliance report example.**

### 3.3.5 Demonstration of the Vocabulary and Compositional Argument Approaches in the Automotive Case Study

Work has been carried out in WPs 4 and 5 to provide tooling and techniques to support argument developers in producing more controlled, structured argumentation to support the assurance of safety-critical systems. The facilities provided are as follows:

- Controlled vocabulary techniques. Assurance cases are subject to a variety of problems of interpretation, consistency and comprehension which result from the flexibility of the natural language medium generally used to express them. Deliverables D5.3 and D4.4 (especially version 1) discuss these problems at length. The OPENCOSS vocabulary approach and tooling provides a methodology to allow for tighter control of the language used to express assurance arguments, so that arguments can be clearly understood by their various target readerships (who are typically spread across different countries, organisations and concerns (i.e. certification authorities, lawyers, engineers). In OPENCOSS, structured vocabularies are produced, to provide unambiguous definitions of key terminology from the domain. These vocabularies are then used to provide the basic lexicon used to develop the arguments. For the guidance of the users, the vocabulary tooling provides autofill suggestions and dictionary lookup functions, integrated with the Argument Editor tool. In order to avoid issues with syntactic ambiguity (which might, for example, cause confusion as to the precise referent or scope of an argument claim), a methodology is provided to allow the development of an argument using structured expressions to capture claims about particular
aspects of the system and its assurance. The structured vocabulary can be used to provide the user with suggestions as to potential terms with which the claim expressions can be instantiated.

- Compositional argumentation approaches. Typically, assurance arguments for large-scale safety-critical systems are not developed monolithically, but represent an integration of argumentation and data relating to a number of concerns. For example, a typical approach (and that envisaged most commonly in OPENCOSS) is for a system-level assurance case to be developed with a number of assurance cases encapsulating the data for the system’s constituent components. This reflects the development process now common for safety-critical systems, which tend to integrate numerous constituent, standalone parts. It also represents good engineering practice, assisting in the management of change for arguments, the separation of concerns and – where the supply chain requires it – the privacy of data. OPENCOSS has therefore developed methodology and tooling to support the development of compositional arguments. This includes support for the development of arguments in a modular form (encapsulating data) and for a contract-based approach, based on rely-guarantee relationships between argument modules. Tooling is provided to auto-generate some of the rely-guarantee contracts between argument modules (in a component-based approach), and also to provide for the manual addition of additional argumentation where there is an incomplete match between the requirements of the requiring module and the provision guaranteed from the supplier. The modular approach is described in detail in D5.3, D5.4, D5.5 and D5.6.

3.3.5.1 Vocabulary Creation

OPENCOSS has developed a Vocabulary Editor to support the development of structured vocabularies to control the use of English in argument models (and potentially in other OPENCOSS models). This Editor implements the Vocabulary Metamodel in D4.4, and is described in D4.6.

The Vocabulary Editor has been used in the Automotive Case Study to define vocabulary models for the automotive domain. The models are defined at two levels: a large model (> 300 entries) has been defined to capture the vocabulary of the ISO 26262 standard, and a smaller model (< 50 entries) has been defined to capture the project-level vocabulary used in the case study argument example. The project-level vocabulary is a partial instantiation of the standard vocabulary, with project-specific instantiations mapped to the category structure and terms of the standard-level vocabulary. Figure 26 and Figure 27 show screenshots of the ISO 26262 vocabulary model in the Vocabulary Editor.
Figure 26. Screenshot of the Vocabulary Model for ISO 26262, focusing on the general categories which are used to structure the vocabulary.

Figure 27. Screenshot of the Vocabulary Model for ISO26262, illustrating the modelling of terms.

Figure 26 illustrates part of the category structure which is used to provide the overall structure for the vocabulary. In order to provide a loose hierarchical structure, the Categories are modelled as
subcategories of high-level categories, such as “Artefact”, in many cases mapping to the base classes of the CCL Metamodel. Layers of subcategories are then indicated: for “Artefact”, these include “Document”, which then has subcategories including “Report”. This category structure is indicated in Figure 26 by the category properties pane indicated in the top-left pane of the Editor. The example displayed here is for the “Artefact” category. The attribute “Subcategories” indicates the cluster of categories which are modelled as subcategories of “Artefact”, and the “Terms” attribute lists the terms which are modelled as belonging to the general category.

Figure 27 illustrates some of the terms which are modelled in the 26262 vocabulary. The top-left pane again shows the attributes of the term, in this case the example is “E/E technology”. The controlled English definition of the term is shown in the “Definitions” field – here, the meaning of the term is captured in terms of other vocabulary entries. The definition refers to other items in the vocabulary using the label “voc:” in this example. The look-up function described in the next section allow for cross-comparison of the definitions of different terms in chains of reference, such that the meaning of a given term can be unequivocally identified. The “is-a” and “has-a” attributes allow for the finer-grained structuring of the vocabulary model, based on relationships between terms. The “synonyms” field allows for alternative forms of the term, or for terms with an identical meaning, to be recorded. A free-form “Notes” field allows for further information concerning the usage of a term to be recorded.

One of the principal concerns with the creation of the case study vocabulary was the scale of the task. For reasons discussed in D4.6, providing precise definitions for terms, based on their usage within a standard, a project or a company is perforce a manual task: it requires considerable engineering judgement and linguistic knowledge to establish which are significant terms, how they can be categorised and related to one another, and to disambiguate the characteristically inconsistent usage of terms within the existing standards. This is not solely an issue in the Automotive domain, but is an inevitable overhead of natural language documentation. The task is time-consuming: development of the standard-level vocabulary for ISO 26262, which in its fullest, but still incomplete version contains over 300 individual terms and some 40 categories, took more than six weeks’ effort. However, the task of term extraction was considerably simplified by the use of techniques from Computational Linguistics. Specifically, the developer used inbuilt functions in the Python language and the resources of the Natural Language Toolkit (www.nltk.org) to identify the most frequently-used terms in the entire 26262 standard (a frequency distribution graph was prepared for each section of the standard, and the top 200 nouns from each section were selected), the 100 longest words in each section and all hapax legomena (words which occur only once in the text). Taken together, these three measures are generally held by linguists to provide a reasonable characterisation of the terminology used in a given text. Once the duplicates in each of the lists were removed, the terminology set extracted by these techniques demonstrated considerable homogeneity of usage across the 9 sections of the standard which were analysed: a total list of 624 candidate terms was produced. These were then used as the input to the manual development of the 26262 vocabulary.

3.3.5.2 Compositional Argument Development

Data from the automotive use case was used to validate the support for the development of modular argumentation provided in the Argumentation Editor. The compositional argumentation approach, which is reported in D4.4, D5.3, D5.4, D5.5 and D5.6, is a major contribution from the OPENCOSS project, since it extends existing metamodels of structured argumentation, most notably the OMG’s SACM, to allow for the modularisation of argument strands. The Automotive case study was chosen to demonstrate the Compositional Argumentation approach because the example – the ePark component – is developed as a SEooC, and therefore falls naturally into a compositional development structure, where a system is developed as an integration of multiple discreet, bounded components.

The fact that the ePark component is developed as a SEooC did, however, provide some challenges to the demonstration of the compositional approach. The OPENCOSS approach – in common with other modular approaches to assurance, most notably the Industrial Avionics Working Group’s (IAWG) approach and argument-based approaches for IMA and Autosar (and indeed the SEooC concept itself) - relies on the
The overall modular structure of the argument is captured in Figure 28. Arrows represent dependencies between the modules.

The modules developed in the argument are as follows:

- **eParkComponent (Figure 29)**: This module carries the main burden of the argument. It provides argumentation to support the high-level claim that the ePark item is acceptably safe to operate in its defined context. The claim is supported by argumentation to the effect that all credible failure modes are acceptably mitigated in the design and operation of the ePark item. Failure modes are identified from the Preliminary Hazard Analysis (HARA) data which was provided by the case study. The claim is then made that the Functional Safety Requirements identify the total functionality required to mitigate the defined failure modes. The argument proceeds to take one function
defined by a safety requirement (that the ePark item will maintain the park_pawl disengaged when an incorrect engagement command has been given), and break it down into its constituent subfunctions. Claims about the subfunctions are made in the context of reliability of performance which can be used to bolster the overall claim about the performance of the function. Note that this reliability data is invented, and has been included to demonstrate the contract-based approach to argument composition. In several cases, the lower-level subfunctions rely on the performance of functionality by peer-level components in the system – the Vehicle Control Unit (VCU), Power Supply and the Gear Selector Module (GSM). In these cases, the eParkComponent example comprises a series of claims requiring functionality from the peer-level components at particular reliability and availability. These are matched by claims in the modules relating to the peer-level components, which provide guarantees of services. In cases where the map between the requirement and the guarantee is exact, an away goal reference is used in the source module (eParkComponent) to point to the goal providing the guarantee. Where the map is not exact, an a contract module is provided, containing an argument demonstrating the acceptability of the guarantee or recording the re-negotiation of the requirement.

- **FSR_Argument (Figure 30):** This module records an argument to support the claim that the Functional SafetyRequirements defined for the ePark item are sufficient to cover all credible failure modes. The argument is partially developed in the example here, since insufficient data was available in the case study, but claims concerning the use of recognised requirements analysis techniques to identify the requirements, the use of suitably qualified personnel to carry out the argument and agreement between the allocation of requirements to components and the overall system architecture and objectives are identified.

- **PowerSupply_Argument (Figure 31):** This module records an argument concerning the availability of power from the power supply to all vehicle-level systems. The argument is essentially black-box, with only claims relating to the supply of power to the ePark item required for the example.

- **Contract Module to Resolve eParkComponent.G9 (Figure 32):** This module contains the contract arguing that the ePark requirement for power supply at a particular level of availability (which is specified in Goal G9 of the eParkComponent module) is adequately met by the power and reliability guarantee provided in the PowerSupply_Argument (GPS3). As will be discussed below, the contract is partly autogenereted and partly manually augmented.

- **VCU_Argument (Figure 33).** This module contains an argument to the effect that the VCU_Component performs all of its required functionality (which is concerned with message passing between the GSM and the ePark Control Unit) to a given level of reliability. The argument is broken down over all required functionality. Once again, the strategy is not fully developed, since only the functionality relating to the ePark is included in the module for the case study. This module supplies guarantees of functionality to satisfy (or satisfice) the requirements in the eParkComponent module.

- **GSM_Argument (Figure 34).** This module contains an argument to justify a high-level claim about the reliability with which the GSM supplies position data and command data to the VCU. It is used to provide goals to which away goals in the VCU_Argument module refer.

- **Contract Module to Resolve eParkComponent.G6 (Figure 35):** This module contains the contract arguing that the ePark requirement (in goal G6) for the VCU component to identify incorrect commands input from the GSM component with a target reliability is satisfied, or satisficed\(^\text{1}\), adequately by the guaranteed performance supplied by the VCU component.

- **ContractModule to Resolve eParkComponent.G7 (Figure 36):** This module contains the contract arguing that the ePark requirement (in goal G7) for the VCU component to transmit commands

---

1 “Satisficement” is a term used in Requirements Engineering to capture situations where lower-level requirements do not exactly provide the services/functionality/data required in a higher-level requirement, but the variation in the level of provision they do make are acceptable. Since the match is not exact, the term “satisfaction” is not appropriate, hence “satisficement”,
data to the ePark Control Unit with a target reliability is satisfied, or satisficed, adequately by the guaranteed performance supplied by the VCU component.

Screenshots of the Argument modules in the editor are presented below:

![Diagram of Argument modules]

Figure 29. EPark Component Argument Module.
Figure 30. FSR_Argument Module.

Figure 31. PowerSupply_Argument Module.
Implementation of use cases on top of OPENCOSS platform

Figure 32. Contract Module to Resolve eParkComponent.G9.

Figure 33. VCU_Argument Module.
Figure 34. GSM_Argument Module.

Figure 35. Contract Module to Resolve eParkComponent.G6.
The argument models for the modular argument have been developed with the assistance of the vocabulary tooling provided in WP5, which are reported in detail in D5.5 and D5.6. These tools provide the following functionalities:

- Suggested autocompletion of text using reserved terms defined in the 26262 vocabulary
- Term highlighting – as will be seen in the preceding diagrams, reserved terms (terms which are included as term entries in the 26262 or project vocabularies) are highlighted when they occur in claims
- Definition lookup – the definitions of reserved terms can be displayed by hovering a mouse over a highlighted term in the argument editor. This helps to ensure that terminology is used and understood in a consistent way throughout the modular argument structure.
- Support for claim types. As discussed in D5.3 and D5.6, the category structure of the vocabulary can be used to supply informed guidance on the instantiation of typed structured expressions, which are used to govern the syntax of commonly-occurring claims in assurance arguments, providing patterns for the expression of claims and helping ensure consistency and clarity of scope. The following figure shows the use of the Claim Types Editor tool to specify examples of the types of claim structures which may be patterned for regular reuse. References to the categories used to structure the vocabulary are supplied in the claim structures using {}. In the Argument Editor, a claim type (pattern) can optionally be selected for a given claim. The user is then presented with drop-down lists containing all of the terms assigned to the relevant categories, from which he can select to instantiate the argument.

Figure 36. Contract Module to Resolve eParkComponent.G7.
3.3.5.3 Generation of Argument Contracts

Figure 32, Figure 35 and Figure 36 above represent contract modules used to record agreements between argument modules. Contracts are required in rely-guarantee-based compositional arguments, to record the justification as to why a guarantee (typically of some level of functionality or data supply) made in an argument module relating to one component can be accepted as adequate fulfilment of a requirement for that functionality in another module relating to another component, even though the guaranteed level of service does not absolutely match that stated in the requirement. A contract module records the process of agreement. In cases where the requirement matches the guarantee completely, without the need for renegotiation of the requirement, an away goal reference is typically used to join the two argument modules, so that the logic of the overall argument can be followed easily.

WP5 has provided tooling to allow for the automated generation of some argument contracts. The tooling is described in detail in D5.6. The tool allows for the automated instantiation of a high-level contract argument pattern to deal with one important aspect of modular argumentation. The tooling first takes from the requiring and supplying modules the claims in which the original requirement and guarantee are stated. The tool then instantiates the first part of the argument which links the two, essentially supporting an “acceptability” claim concerning the guarantee. The generated pattern deals with one important aspect of the argumentation approach. This is the fact that claims are valid only in the context in which they are originally stated - claims in both the requiring and supplying modules are constrained by a lot of contextual information, represented in GSN by assumptions, justifications and contexts. An important part of the contract module is therefore to demonstrate that the context in which the original claims were made matches. A pattern which can be used to make such an argument is presented in (Hawkins and Fenn, 2007 – see D5.6 for further discussion). The Contract Tooling takes from the requiring and supplying modules all of the context, justifications and assumptions which are associated with the requirement and guarantee goals. In assurance arguments, all goals descending from (i.e. below in the argument hierarchy) a goal which is stated in a particular context are taken as being in the scope of that context, unless stated otherwise. The tool therefore also uses all higher-level contexts, assumptions and justifications (together
with the goals to which they are attached) and any relevant goals and contexts below, in order to provide input for the argument pattern concerning the similarity of the contextual understanding. An instantiation of the contract pattern is then generated. Note that the argument about context may not alone provide a sufficient argument for the acceptability of the “trade-off” undertaken between requirement and guarantee. Further argumentation may be required, for example to record the process of agreement for the weakening of a requirement. The Contract Tooling allows for manual editing of the generated contracts to supply additional argumentation where required.

The following figures show screenshots from the Contract Tool:

![Figure 38. Selecting Goals to be Resolved by Contract.](image)
It should be noted that the Contract Tooling developed in WP5, as described in D5.6 also provides functionality to instantiate system integration argument patterns (concerning the integration of a component argument into a system-level argument) and to validate contracts. The functionalities have not been demonstrated in the Automotive case study, since the data provided contained insufficient information concerning the system (i.e. vehicle) into which the ePark component is to be integrated.
3.4 General conclusion and improvement

Automotive current projects are mainly manually managed and assessed. OPENCOSS platform features have highlighted the way of modeling the automotive reference standard ISO 26262 and its application for generating the project baseline and safety case. The ISO 26262 standard is also novel and has never been represented until now in a suitable semi-formal visual language as that provided by CCL into the platform. The platform features, in particular, improve the management of the artefacts (argument and evidences) stored into the project safety case, allowing a way to control and the assessment of the process during and after its course.

The OPENCOSS platform can be improved in some graphical features and data access management.
4 Avionics Case Study

This section presents an implementation report of the Avionics Case Study as it was deployed to benchmark OPENCOSS approach and tools.

4.1 Case Study Specification

The OPENCOSS deliverable D1.2 describes the Avionics Case Study. It corresponds to a situation of reuse of an execution platform (computing unit and operating system) developed in the railway domain and to be certified in the avionics domain. The goal is to build the qualification dossier, based on assets and artefacts provided with the reused parts, without or limiting the rework. The qualification dossier is then presented for certification.

The execution platform is considered as an independent item for which a qualification dossier will be built. This qualification dossier consists of plans, technical documents, and certification documents. Technical documents are specifications, validation and verification life cycle data. The certification documents are configuration index documents and accomplishment summaries.

The initial execution platform and the associated documentation issued from the railway domain comply with railway standards (CENELEC EN50128). The final execution platform and the elaborated qualification documentation to be used in avionics domain must comply with avionics standards (ED-12c/DO-178).

4.2 Case Study Source Data

4.2.1 Railway Source Data

Life cycle data used from the railway domain are described by the figure below in the Table hereafter. In the scope of this study, the full railways project document references are provided in the OPENCOSS tool and listed hereunder. Document entries in OPENCOSS tool are used to indicate how railways artefacts comply with the EN50128 standard and will be used by the avionics project as equivalent artefacts.
### Table 1 Railway Life Cycle Data

<table>
<thead>
<tr>
<th>Document Name Identified in Opencoss Tool</th>
<th>Document Reference</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Management Plan</td>
<td>3CR-00000-6107-DPAPA</td>
<td>1</td>
</tr>
<tr>
<td>Configuration Management Plan</td>
<td>3CR-00000-5260-QMAPA</td>
<td>2</td>
</tr>
<tr>
<td>Documentation Plan</td>
<td>3CR-00000-5034-DCAPA</td>
<td>9</td>
</tr>
<tr>
<td>Quality Management Plan</td>
<td>3CR-00000-5871-QRAPA</td>
<td>1</td>
</tr>
<tr>
<td>Quality Management Plan Report</td>
<td>3CR-00000-5872-QRAPA</td>
<td>2</td>
</tr>
<tr>
<td>Coding Rules for C</td>
<td>3CR-00000-5018-QRAPA</td>
<td>3</td>
</tr>
<tr>
<td>Software Techniques and Measures</td>
<td>3CR-00000-5069-DUAPA</td>
<td>3</td>
</tr>
<tr>
<td>Application relevant problem reports</td>
<td>3CR-00000-5869-DUAPA</td>
<td>6</td>
</tr>
<tr>
<td>Release Notes</td>
<td>3BY-00000-AAAA-FMAPA</td>
<td>5</td>
</tr>
<tr>
<td>System Requirements Specification</td>
<td>3CR-00000-5255-DTAPA</td>
<td>5</td>
</tr>
<tr>
<td>System Requirements Specification Safety</td>
<td>3CR-00000-5275-DTAPA</td>
<td>4</td>
</tr>
<tr>
<td>Feature List</td>
<td>3CR-00000-5215-FLAPA</td>
<td>7</td>
</tr>
<tr>
<td>Requirements Traceability Matrix</td>
<td>3CR-00000-5278-DHAPA</td>
<td>3</td>
</tr>
<tr>
<td>Subsystem Requirements Safety</td>
<td>3CR-00000-5413-DTAPA</td>
<td>4</td>
</tr>
<tr>
<td>Subsystem Requirements OS</td>
<td>3CR-00000-5085-DTAPA</td>
<td>9</td>
</tr>
<tr>
<td>Top Level Architecture</td>
<td>3CR-00000-5030-DEAPA</td>
<td>2</td>
</tr>
<tr>
<td>Detailed Design Document OS</td>
<td>3CR-00000-5094-DSAPA</td>
<td>4</td>
</tr>
<tr>
<td>Detailed Design Document Compact Flash</td>
<td>3CR-00000-5350-DSAPA</td>
<td>3</td>
</tr>
<tr>
<td>Handbook of Methods</td>
<td>3CR-00000-5038-PCAPA</td>
<td>2</td>
</tr>
<tr>
<td>Verification Plan</td>
<td>3CR-00000-5025-DUAPA</td>
<td>4</td>
</tr>
<tr>
<td>Verification Report</td>
<td>3CR-00000-5571-QZAFA</td>
<td>2</td>
</tr>
<tr>
<td>Test Report OS</td>
<td>3BY-00000-0001-QZAPA</td>
<td>2</td>
</tr>
<tr>
<td>Test Design Specification – System Level</td>
<td>3CR-00000-5128-QPAPA</td>
<td>6</td>
</tr>
<tr>
<td>Test Design Specification – OS Level</td>
<td>3CR-00000-5317-QPAPA</td>
<td>6</td>
</tr>
<tr>
<td>Intake Report – OS</td>
<td>3CR-00000-5592-DFAPA</td>
<td>3</td>
</tr>
<tr>
<td>Validation Plan</td>
<td>3CR-00000-5485-DUAFQ</td>
<td>1</td>
</tr>
<tr>
<td>Validation Report</td>
<td>3CR-00000-6004-QZAFA</td>
<td>1</td>
</tr>
<tr>
<td>Validation of Requirements</td>
<td>3CR-00000-6404-QZAPA</td>
<td>1</td>
</tr>
<tr>
<td>Validation report C90</td>
<td>3CR-00000-5572-QZAPA</td>
<td>1</td>
</tr>
</tbody>
</table>
4.2.2 Avionics Source Data

Life cycle data used from the avionics domain are described by the figure below in the Table hereafter. Those documents were mainly used to provide evidences while creating the DO-178C model and illustrating the typical artefacts used in an avionics project. Those artefacts answer to both DO-178C standard and internal company referential.

<table>
<thead>
<tr>
<th>Table 2 Avionics Life Cycle Data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Document</strong></td>
</tr>
<tr>
<td>Checklist_HLR.xls</td>
</tr>
<tr>
<td>Checklist_LLR.xls</td>
</tr>
<tr>
<td>Opencoss_PAS_Notebook.doc</td>
</tr>
<tr>
<td>Opencoss_PRS.doc</td>
</tr>
<tr>
<td>Opencoss_PSAC.doc</td>
</tr>
<tr>
<td>Opencoss_SAS.doc</td>
</tr>
<tr>
<td>UD_Examples.doc</td>
</tr>
<tr>
<td>Eurocae ED-12C</td>
</tr>
<tr>
<td>Eurocae ED-124</td>
</tr>
<tr>
<td>SAE ARP 4754A</td>
</tr>
</tbody>
</table>

![Figure 41 Avionics Life Cycle Data](image-url)
4.3 Case Study Implementation in OPENCOSS

The global cross domain project structure from system level to software level allows considering a dual use of the case study, the classical way of certification and the incremental way of certification.

![Diagram: Dual use of the case study](image)

4.3.1 Avionics and Railway Safety Standard Reference Framework Modeling

4.3.1.1 Conceptual Modeling of Safety Standards and Capture Information from Ref. Framework

Information Source used are ED-12/DO-178, ED-124/DO-297, SAE ARP 4754A and CENELEC EN50126, EN50129 and EN50128.

Due to the reuse objectives from railway domain to avionics domain, it is necessary to analyse the standards multiple angles or views.

4.3.1.1.1 Safety Standards Documents Framework

The structural analysis of the standards using the main subjects described by the table of content of each of them can identify the macro objectives addressed.
As showed by the figure above, standards have some discrepancies, among which:

- Roles and responsibility (§5 of EN50128) in railway are no equivalent in avionic,
- Validation in avionic is an Aircraft/System dedicated process and a part of ED79A/ARP4754A,
- Generic Software Development (§7 of EN50128) in railway are no equivalent in avionic at DO-178 level. Therefore, at system level, the Technical Standard Order (TSO) may be viewed as a generic development regarding the targeted aircraft but with the intended function well specified,
- Software deployment and maintenance (§9 of EN50128) in railway are no equivalent in avionic at DO-178 level. Therefore, at system level, the means of compliance of Certification Specification 25.1529 “Continued Airworthiness” may be viewed as an equivalent objective.

4.3.1.1.2 Safety Standards Concept comparisons

The analysis of the definitions provided by the standards can identify the concepts used and addressed.
As showed by the figure above, standards have some discrepancies in term of definition, among which:

- **Safety function** in railway is the equivalent of avionic safety-related\(^2\) functions at A/C definition level.
- **Validation** in avionic is an Aircraft/System dedicated process and a part of ED79A/ARP4754A.
- **Transition criteria** are an important asset for avionic domain, based on process control demonstration.
- **Derived Requirement** is an important asset for avionic domain, based on intended function demonstration (Certification Specification 25.1301).

### 4.3.1.1.3 Safety Standards strategy comparisons

The analysis of the strategy provided by the standards can identify the *intent of the objectives* addressed.

---

\(^2\) * Safety-related as defined inside “Model-Based Assurance of Safety-Critical Product Lines” – Phd Thesis Habli – 2009 - University of York
As showed by the figure above, standards have some discrepancies in term of strategy:

- **Prescriptive Product Based** for railway domain, Specific design features are required and generic product oriented
- **Prescriptive Process Based** for avionic domain, the standards specify the process to be used in producing the specific product or specific system

A general model for avionics safety standard can be provided by similarity with a control loop, double and reversed, Figure below, which enables to oversee the quality of the design process, including the achievement of the intended function and evidence of compliance associated.

---

**Figure 45 Safety Standards Strategy analysis**

**Figure 46 Avionic Safety Standard based on control loop principle, double and reversed**

4.3.1.2 Implementation of EN50128 into the Opencoss tool

---

3 • as defined by N.G.LEVESON - MIT - Journal of System Safety Vol. 47, No. 6 • November-December 2011 “The Use of Safety Cases in Certification and Regulation”
The figure below shows how the railways EN50128 standard has been effectively implemented into the OpenCoss tool, using the Common Certification Language: sections of the document have been modeled into Objectives and the requirements sub-sections into requirements. Documents expected as part of Table A of EN50128, are modeled into “Artefacts” in the OpenCoss tool.

**Figure 47:** OPENCOSS Tool: Tree view of EN50128 model
4.3.1.3 Implementation of DO178C into the Opencoss tool

As per the railways standard, the avionics DO-178C standard has also been modeled according to a similar method. The figure below show how the DO-178C standard life cycle data has been effectively implemented into the Opencoss tool, and how artefacts in the tool are linked to their constraining requirements.

Adding constraining requirements to an artefact will help in building the compliance mapping between the project artefacts and the standard requirements.

Figure 48: Opencoss Tool: Tree view of DO178C model, focused on artefacts and their constraining requirements
4.3.1.4 Railway and Avionics Framework Validation
Validation of standard has been done by an internal assessor of the company. The aim of the validation is to assess that the modelling of the avionics standard is the correct model and that it is complete regarding the use case boundary.

4.3.2 Equivalence Mapping Modeling
Previous chapter explain the first step for the analysis of the cross domain safety standards. The need in term of cross domain reuse is to provide a means to claim the equivalent level of safety from a domain to another and to identifying the remaining activities (for partial or no equivalence) to do from one domain to another, in our case, from railway domain to avionic domain.

![Figure 49 Step 1 : Standard appropriation](image)
This need can be met by building an equivalence mapping from a safety standard to another.

![Figure 50 Equivalence mapping approach](image)
We suggest two others steps, as described by Figure 50 above:

**Step 2** - Prescriptive Product Based objectives to equivalent Prescriptive Process Based objectives analysis
Step 3 - Cross domain: transfer function analysis Prescriptive Cross Domain Based ➔ PXB
- traceability elaboration,
- orphan identification and
- completeness analysis

Figure 52 Completeness toward common performance based abstraction
The figure above shows the actual implementation in the Opencoss tool, the equivalences between models. In the example given above, the equivalence is done at requirement level, between an avionics DO178C requirement to two EN50128 requirements (here 7.2.4.13 and 7.2.4.14).

By completing the model with those equivalence links, we are now able to link the DO178C artefacts to the railways EN50128 requirements.

**Figure 53 Equivalence Mapping in Opencoss tool**
4.3.3 Company Project Modeling

4.3.3.1 Conceptual Modeling of Company Project and Capture Information from Ref. Framework

Regarding preliminary safety standard analysis, Cross Domain Use Case concept shall satisfy three constraints expressed in the form of requirements

- Associated Process and Design Assurance, Process domain shall be reusable from source domain to target domain,
- Technical Solution, Design details shall be available from source domain to target domain and without specific technics,
- Intended function boundary, the selected Intended function shall be reusable from source domain to target domain.

4.3.3.2 Map Information from different Ref. Framework

4.3.3.2.1 Process source domain regarding to the Process target domain

Thanks to common Thales Quality System at Thales group Level, it is easier to identify the specificity of each safety standard from source to target domain for the use case. The technical reference provide by the Quality system is showed below, it facilitates the comparison of the two domains providing a means of interpreting the solutions lifecycle data.

```
<table>
<thead>
<tr>
<th>Railway</th>
<th>Process View</th>
<th>Life Cycle Data View</th>
<th>Avionics</th>
<th>Life Cycle Data View</th>
<th>Process View</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDP</td>
<td>SRS</td>
<td>Equivalent ~90%</td>
<td>SRS</td>
<td>SDP</td>
<td></td>
</tr>
<tr>
<td>QAP</td>
<td>SDD</td>
<td>Partially ~50%</td>
<td>SDD inc. Ctrl &amp; Data flow description</td>
<td>QAP</td>
<td></td>
</tr>
<tr>
<td>SEMP</td>
<td>Code</td>
<td>Partially ~40%</td>
<td>Code</td>
<td>SEMP</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>Executable Object Code</td>
<td>Different 0%</td>
<td>Executable Object Code</td>
<td>PSAC</td>
<td></td>
</tr>
</tbody>
</table>
| ...             |              |                                       | SAS           | ...
```

The concepts of equivalence set {“Equivalent”, “Partially”, “Different”) address structural aspects and not compliance of the lifecycle data regarding the safety standards.

4.3.3.2.2 Design lifecycle data from source domain

For Cross domain Use Case, Technical Solution shall be in the scope of System features (Mass Memory management,…) or SW utilities (Math Library, Files services,…), the selection process has been submitted to a decision tree similar to the one below
4.3.4 Reuse of railways artefacts in avionics domain

4.3.4.1 Using the opencoss tool to match railways artefacts to DO178C requirements

Thanks to the preliminary modelling of the DO178C and EN50128 standards, using the Common Certification Language, artefacts and their constraining requirements are setup in the Opencoss tool. A first step for the use case is to complete the implementation of the railways project in order to build a full compliance map linking the evidences (life cycle data) of the project to the EN50128 standard requirements.

Once the compliance map is built on railways side, an avionics project may be created, referring to the railways project and using its life cycle data through the equivalence map that exists between DO-178C and EN5128 standard. Having an equivalence between railways standard requirement and avionics standard requirement does not automatically imply there is no need for a manual check, but this feature of the Opencoss tool greatly helps in building a “pre-compliance” framework as shown below.
4.3.4.2 Creating the railways project in the OPENCOSS tool

The railways project is built in two phases:

- Evidences are integrated into the evidence folder so that all applicable life cycle data are identified in the tool as a reference for compliance to the standard requirements
- Compliance Map is built for each standard requirement with an evidence found in one or more artefacts created in the first step.

![Figure 55 Mapping EN50128 requirements to evidences](image1)

![Figure 56 Identifying evidences compliance map](image2)
4.3.4.3 Creating the avionics project in the OPENCOSS tool

The avionics project may be built in several ways. An automated way is provided by the OPENCOSS tool, in order to create a typical DO178C life cycle data evidence folder; or artefacts may be added manually to match the internal company life cycle data format (i.e. some project document may be the merge of several DO178C standard documents).

In order to build this use case, we chose the automated evidence tree build in order to focus on the reuse aspects of the railways project for compliance in the avionics domain. Here, the key features of the tool are the equivalence map between DO178C and EN50128 standards and compliance map between EN50128 standard and the project evidences.

Using the “Cross Domain” functionality of the Opencoss tool (see next page), avionics project members can easily chose which railway evidence may be mapped to the DO178C requirements.

In the example below, the left side of the screen represents the DO178C project tree with the standard requirements on the top-left side and the automatically-built life cycle data on the down-left side. The highlighted requirement 11.9b is one of the constraining requirements of the “Software requirements Data” document. This requirement is mapped to the EN50128 7.2.4.7 and 7.2.4.8 requirements highlighted in green; so compliant evidence in railways project to those requirements are bound to fulfil the DO178C 11.9b requirements.
Figure 57 Building Compliance Map to DO178C using the Cross Domain Opencoss tool feature
4.4 General conclusion and improvement

Avionics use case has been fully manually assessed in a real project, OPENCOSS platform feature have highlight the matching the railways artefacts to the DO178C requirements, in preparation of the reuse of railways artefacts.

In conclusion, the CCL certification specific language is very helpful to improve understanding of crossed prescriptive knowledge and the associated standards, and, the method of building an equivalence mapping between standard is very helpful to assess cross domain objectives for internal assessor and with legal assessors. The OPENCOSS platform may be improved regarding to industrial efficiency need (mainly about data access and workstation deployment) and robustness to the use.
5 Railway Case Study

This section presents an implementation report of the Railway Case Study as it was deployed to benchmark OPENCOSS approach and tools.

5.1 Case Study Specification

The OPENCOSS deliverable D1.2 describes the Railway Case Study. It corresponds to the certification of a railway signalling system. It describes the certification of a European standardized signalling system provided by Alstom Transport. The case study is related to the GATC subsystem of the Alstom GATC solution for the ETCS of the ERTMS, known as EVC in the architecture of ERTMS.

The GATC trainborne subsystem is Alstom’s generic solution for ETCS on-board equipment that will be used by Alstom ERTMS application projects. The main functions of this subsystem are to ensure safe movement of the train and to inform the driver by means of a cab display facility. This case study is a generic sub-system that is parameterized for specific project application. This generic sub-system contains also railway generic products.

The GATC trainborne is an existing subsystem and already certified. This case study provides real data to run the OPENCOSS tool platform. The objective is not to re-certify the subsystem but to evaluate the benefits of the model-based certification approach on a real Alstom Transport subsystem. Since this case study is a generic application (EN 50129 standard), the specific application engineering activities are out of the scope of the case study (tailoring of the generic application for a specific project application). The case study will also deal with the reuse of the GATC subsystem from one country to another.

The data provided for the case study contains parts of the documentation that was used to obtain the CENELEC certification of the subsystem. This documentation has been sanitized for intellectual property reasons and is composed of the safety case, the safety plan, the hazard log, and a part of the safety studies (FTAs and FMEAs). The CENELEC standards used for this case study are EN 50126 and EN 50129. Compliance with national rules will also be addressed. In this sense, some safety requirements are coming from specific application projects, and national safety authorities define requirements specific to the country where the generic application may be installed.

It was not planned to release project management information other than the safety plan that describes the entire process of safety related activities. The data required for project management metrics involving workload, costs, etc., would not be released. Therefore, Alstom Transport will perform the evaluation of these metrics internally, and only ratio results will be communicated.

5.2 Case Study Source Data

There are three main sources of information to implement the Alstom Case Study: the CENELEC standards used to measure compliance of the assurance projects, and the two GATC assurance projects, for different countries (for Belgium and for France), used to measure the delta assurance/certification efforts.

5.2.1 CENELEC Standards

We used the following versions of CENELEC standards:

• EN 50126: Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS), September 1999.

• EN 50128: Railway applications - Communications, signalling and processing systems - Software for railway control and protection systems

The European Standard CENELEC 50129 defines the conditions that shall be satisfied so that a safety-related electronic railway system/sub-system/equipment can be accepted as adequately safe for its intended application. The conditions for safety acceptance are presented in this standard under three headings, namely:

• Evidence of quality management,
• Evidence of safety management,
• Evidence of functional and technical safety.

All of these conditions shall be satisfied, at equipment, sub-system and system levels, before the safety related system can be accepted as adequately safe. The documentary evidence that these conditions have been satisfied shall be included in a structured safety justification document, known as the Safety Case. The Safety Case forms part of the overall documentary evidence to be submitted to the relevant safety authority in order to obtain safety approval for a generic product, a class of application or a specific application. For our Railway Case Study the focus is on the Safety Case for the Generic ATC (GATC) trainborne subsystem. The structure of the Safety Case is illustrated in Figure 58.

![Figure 58. Safety Case Structure according to EN 50129](image)

The quality management system shall be applicable throughout the GATC Trainborne Sub-system life-cycle, as defined in EN 50126, as shown in Figure 60.
Figure 59. Example of System-life cycle according to EN 50126

5.2.2 Belgium GATC Assurance Project

Figure 60 illustrates the Safety Assurance activities of the GATC Trainborne Sub-System within the wider context of the ERTMS Application Project. The (yellow) “ERTMS Application Scope of Work” area is given as an example of an Application Project with its typical Safety Management. In practice, each ERTMS Application Project describes its own Safety Management Process in the related Project Safety Plan. The (light blue) “GATC Trainborne Scope of Work” area is the scope of the OPENCOSS case study.
Implementation of use cases on top of OPENCOSS platform

Scope of Work

- Software Development
- Hardware Development

- Safety Analysis (FMECA, ETA, FTA)
- Maintainability Analysis
- Allocation of reliability data
- Preliminary Operational Analysis (HAZOP)
- Update of the Hazard Log & Safety Plan

- RAMS studies during tendering based on relevant data and experience on RAMS coming from operation of other similar equipment
- RAMS standards

Customer's Specifications

- General System Overview
- System Requirements Definition ERTMS Level x
- Specific System Design ERTMS Level x
- Specific Sub-System Requirement Definition and Design
- Sub-System Requirement Specification
- Sub-System Acceptance (Sub-System Validation)
- Sub-System Integration & Acceptance Tests Reports

ERTMS Application Project

- System/Sub-System Scope of Work
- Monitoring, Maintenance and Revision
- System Acceptance and Commissioning (System Validation)
- System Acceptance Test Specification
- Safety Validation Specification
- Sub-System Integration & Acceptance Tests Reports
- System Integration Tests Reports
- System Acceptance Tests Reports
- System Integration Specific System Design ERTMS Level x

Safety Validation

- ERTMS System Safety Case
- Validation Report
- Reliability & Maintainability

GATC trb Support

- GATC trb Safety Case
- (Product) Sub-System Safety Case
- Validation Report

Figure 60. Safety Assurance Activities associated with V-Cycle Phases
The first reference document is the Safety Plan. Figure 61 shows the Table of Content of the Safety Plan at ERTMS application level (Signalling System). The Safety Plan details the methods used to ensure that the safety target (e.g. compliance to CENELEC standards). It introduces also the Signalling Safety Management put in place to satisfy the Safety Assurance requirements. It also includes: identification of independent safety assessment if any, description of the project organization, risk acceptance criteria, safety policy, description of the project, description of the technical system, hazard control strategy (hazard analysis, safety requirements validation, hazard log ...), identification of safety analysis to be performed and reused components.

![Content Table]

Figure 61. Table of Content of the Safety Plan for a ERTMS Application project (Signalling System)

An example of Hazard Log can be seen in Figure 62. This is the core artefact to work on safety-related analyses. The purpose of the Hazard Log is to document and to track all the ATC Trainborne Sub-System Hazards and their related safety mitigation measures from their identification until their elimination or until the associated risk is reduced to an acceptable level.
<table>
<thead>
<tr>
<th>Ref scenario</th>
<th>Description</th>
<th>Subsystem</th>
<th>Resp.</th>
<th>Implementation reference</th>
<th>Safety Case</th>
<th>Test Plan / Installation or Maintenance plan</th>
<th>Related Safety and Verification references</th>
<th>Validation Conclusion</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>RQ_PHA_025</td>
<td>Trackside and on-board ATP must safely determine the ESA (Emergency Stop Area) status, preventing a train by EB application from entering or leaving the station if a PSD is detected unlocked. Vital design of trackside ATP using the platform 2003 and vital design of on-board ATP using coded monoprocessor principles.</td>
<td>ATC Sub-System</td>
<td>ATST</td>
<td>IAR00-20-D400-ATST+0002_ATC FUNCTIONAL SPECIFICATION</td>
<td>20x3 safety case - Y3 - 64 A425246</td>
<td>Hardware (On-board ATP):</td>
<td></td>
<td></td>
<td>Closed</td>
</tr>
</tbody>
</table>

**Figure 62. Excerpts of a Hazard Log for the GATC trainborne subsystem**
The Safety Studies include all the identification of global causation by a deep causal analysis (from elementary faults to system accidents) and to allow analysts to provide explanation of causation link from cause to effect at every levels of the V descendant phase. The principle is to merge the existing safety studies realized at different stage of the descendant phase of the V lifecycle within a global fault tree structure. The global structure is illustrated by the fault tree presented in Figure 63.

In the Preliminary Hazard Analysis (PHA), accidents are identified by the hazardous situation and the operational concept in which they occur; this is the consequence analysis. The table in Figure 64 shows an example of PHA as performed currently in Alstom.
The next step is to work on the SHA (System Hazard Analysis) and SSHA (Subsystem Hazard Analysis). For this, FMEA techniques are used following the next procedure:

**FMEA System (SHA):**
- Identify failure modes for every single system’s function (elementary system safety issues),
- Identify consequences of the failure modes and link them to system safety issues,
- Identify and associate requirements to failure modes: (elementary system safety criteria),
- Identify causes of the failure modes => SSHA;

**FMEA subsystem (SSHA):**
- Identify failure modes for every single subsystem’s function,
- Identify consequences of the failure modes and link them to SHA failure modes causes,
- Identify causes of the failure modes,
- Identify and associate requirements to failure modes;

Figure 65 shows an excerpt of SSHA table as performed in Alstom.

<table>
<thead>
<tr>
<th>Item</th>
<th>Analyzed Function</th>
<th>PBM: Hazard</th>
<th>Cause</th>
<th>Direct effect</th>
<th>Hazardous situation</th>
<th>Potential accident</th>
<th>Safety Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1_1_2</td>
<td>M11: Measure calibrations (mechanics/crushing)</td>
<td>D</td>
<td>Emolence position on the test (more than the real one)</td>
<td>ROS</td>
<td>Train speed and position are underestimated.</td>
<td>Train unable to stop before target point</td>
<td>ATSF</td>
</tr>
<tr>
<td>M1_2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M1_2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 65. SSHA Example of Table

### 5.2.3 France GATC Assurance Project

The second Assurance Project (for a GATC project in France) adds a new feature to the GATC functionality, which is called: “Dynamic configuration of the EVC”. This new feature is “cross country” as now we can switch braking curves on line, as the trains cross the country boundary. Operators have requested the adaptation of the braking curves according to the country where the train is running. Several curves implies aspects to take into account such as service brake vs emergency brake, potential switch from one of another depending on confidence / health of the rolling stock, the status of the line, etc.

To understand the impact of the new feature into the whole GATC trainborne documentation for certification, we use a report of “Safety Assessment of the Dynamic Configuration of the EVC Software”. We can see the Table of Content of such a document in Figure 66. The aim of the report is to verify the safety impact on changing values of parameters dynamically in order to adapt the EVC (our GATC
trainborne subsystem in ERTMS terminology) to the line it has to cover. This limits the change of train in order to adapt a train to a country or a part of the line.

Figure 66. Safety Assessment Report content for the Dynamic Configuration of the EVC

5.3 Case Study Implementation in OPENC OSS

Following the procedures of deliverable D2.4 (Usage Scenarios) for the Railway application domain, we obtained the following results.

5.3.1 Modelling of the EN standards

The first step is to capture the required Railway EN standards to be modelled in the format of OPENC OSS reference framework models.

The EN 50126 standard is the core reference document, which defines the applicable safety-related lifecycle for out GATC trainborne subsystem. Figure 67 shows an excerpt of the main areas to cope with
Implementation of use cases on top of OPENCOSS platform

D1.4

FP7 project # 289011

Page 75 of 96

(information visualized in a structured way), on the basis of the content of the EN 50126 standard (e.g. visualization of the main phases of CENELEC V-cycle). For each of the reference framework concepts, there is a clear structured and textual description of specific compliance requirement of the selected phases (see Figure 68), short text-based note concerning the rationale for each concept, definition of evidences (artefacts) to be created, activities to be performed, roles, development techniques to be followed, etc.

Figure 67. Modeling of EN 50126 (Graphical view)

Figure 68. Example of Reference Requirements for EN 50126 (textual view)
The CEN 50129 standard provides the guidelines to create Safety Cases, and it has been decided not to modeling this standard in OPENCOSS, since the compliance to this standard can be directly considered when looking to the Safety Case structure.

The CEN 50128 standard has been modeled in OPENCOSS, since the software compliance has to be measured regarding this reference document. Figure 69 illustrates some excerpts of this standard following the same procedure as for CEN 50126.

![Figure 69. Excerpts of the modeling of EN 50128 (Graphical View)](image-url)
Figure 70 shows an example of recommendation tables modelled in OPENCOSS.

![Recommendation Tables Modelled in OPENCOSS](image)

5.3.2 Specification of the Assurance Project (Belgium)

The assurance project-specific activities start by building the Belgium GATC trainborne subsystem assurance project in OPENCOSS.

As a first step, we created a new safety assurance project and defined the reference framework, such as the applicable safety standard(s), the name of the project, etc. During the assurance project creation, we tailor the two reference framework (EN 50126 and 50128) into Baseline models which now represents the compliance obligations for the Belgium GATC assurance project (see Figure 71). This assurance project model helps manage the assignment of tasks to organizational units/person(s) and the progress of the tasks’ status (to be done/in progress/done) along the project lifecycle. The OPENCOSS platform provides a way to manage the exchange of information among the organizational units/persons involved in the safety assurance project in an efficient way, making it feasible to get custom feedbacks regarding the current progress status of each specific task and to consequently identify the bottlenecks and delays. For more details on the process management aspects, see Section 5.3.6 of this document.

As a core activity during the assurance project lifecycle it is the evidence management. Evidence Management refers to the characterisation, evaluation, traceability and change analysis of evidential artefacts. The OPENCOSS platform shall show the user the exhaustive set of evidences to be provided to meet the requirements defined by the reference framework. After a given evidential document is created, developed and collected, artefacts must be characterised in terms properties, formats, versions, changes, and lifecycle events. Figure 72 shows an excerpt of the artefact model for the Belgium GATC assurance project.
Figure 71. The Belgium GATC assurance project with the two Baseline models (EN 50126 and 50128)

Figure 72. Excerpt of the Artefact Models for the Belgium GATC assurance project
5.3.3 Safety Case Modelling

The purpose of this Safety Case is to give the evidence that the ATC Trainborne Sub-System has been developed in compliance with the safety acceptance and safety approval requirements, specified in the CENELEC Standard EN 50 129. Figure 73 illustrates the global structure of Safety Case for any ERTMS Application Project. It defines a set of ERTMS Trackside and Trainborne functions that are to be performed by each ERTMS System, as defined in the ATC Project, but the ATC Project provides neither a complete system approach, nor the corresponding System Safety Case, for any Specific ERTMS Project. For the development process, the ATC Project has been cut into two parts: the ATC Trainborne Sub-System and the ATC Trackside Level 2 Sub-System. Our focus is the ATC Trainborne Sub-System safety case.

Figure 73. Global structure of ERTMS Safety Case models (OPENCOSS on top of the Figure)
The Safety Case for the ATC Trainborne Sub-System includes especially the "Quality Management", the "Safety Management" and "Functional and Technical Safety" demonstrations for the sub-system development. We focused on the “Technical Safety Report” section, which provides the risk mitigation argumentation. The "Technical Safety Report" describes the technical principles which ensure the safety of the design.

Figure 74 shows an excerpt of the OPENCOSS argumentation model for the core aspects of the GATC trainborne’s “Technical Safety Report”. Appropriate references are given to the Safety Case report, which provide additional information about the design principles and calculations, the test specifications and results and the safety analyses. This Safety Case model covers the following topics in compliance with the CENELEC Standard 50129:

- G2: Functional behaviour safety of GATC in absence of failures
- G3: Safety assurance of GATC (not a focus of the current case study)
- G4: Mitigation of effects of faults

Figure 74. Safety Case model: GATC trainborne subsystem is adequately safe

G2 is demonstrated by compliance of GATC trainborne requirements, as well as by the assurance of correct hardware and software functionality (see Figure 75). Our focus in the Alstom Case Study is G4.

The purpose of G4 is to demonstrate that the GATC trainborne subsystem continues to meet its specified safety requirements, including the quantified safety target, in the event of a random hardware failure and, as far as reasonably practicable, systematic faults (G4.1).

Figure 75 shows the argumentation for the mitigation of single faults and multiple random faults. The demonstration that the ATC Trainborne Sub-System continues to meet the specified safety requirements in the event of single random hardware failures, recognised as possible, is provided in the trainborne Safety Analyses (G4.1.1), using:

- The FMEA bottom-up method: this method enables to identify the single random hardware failures, and to evaluate their impact on the trainborne sub-system and on the system safety.
- The FTA top-down method supported by the FMEA: the FTA methodology enables to identify the different combinations of the EVC Platform failures which can lead to an unsafe situation.

G4.1.2 demonstrates that single faults of the GATC trainborne subsystem are detected and a safe state is enforced in a time sufficiently short to fulfil the specified quantitative safety target for the GAT trainborne subsystem.

As the GATC subsystem is based on a redundant two-out-of-three architecture, a simultaneous malfunction of two corresponding redundant elements could be hazardous. Demonstration of mitigation of multiple random faults is done by demonstrating the independence between the three processing channels and the independence between redundant elements of the GATC subsystem. G4.2 in Figure 75 provides the argumentation for this aspect.
Figure 75. Safety Case model: GATC behaviour in absence of failures is safe & GATC faults are sufficiently mitigated
Figure 76. Safety Case model: The Hazards are Sufficiently Mitigated
Figure 77. Safety Case model: Defence against systematic failures in GATC is adequate
Figure 76 shows the argumentation for hazard mitigation based on FTA. For each hazard, a FTA has been built, and the results are presented and discussed in the GATC Trainborne Fault Tree Analysis report. A step by step process is put in place in order to build the FTA. That means that when building the FTA, at first the Fault Tree is built taking into account as basic events the boards and assuming that the total failure rate of the boards corresponds to the wrong side failure rate. When this level of decomposition does not allow to meet the target, a sensitive analysis is performed in order to determine which boards have the main contribution and the related boards are decomposed in blocks. The same process is performed at the block level if necessary until the targets are met.

The argumentation also demonstrates that the GATC Trainborne Sub-System fulfils the quantitative safety targets for all UNISIG Top Events (G4.3).

The demonstration of fulfilment of the quantitative safety targets is presented in G4.3.1 to G4.3.4. In summary:

- TE 1-Emergency Brake Application WSF: compliant;
- TE 2-Balise Detection WSF: compliant;
- TE 3-Increased Sensitivity WSF: compliant;
- TE 4-Increased Telepowering Capacity WSF: compliant.

Figure 77 provides arguments related to defence against risks of systematic faults (G4.5). The set of arguments describes:

- Compliance of the GATC Trainborne Sub-System (EVC Platform) architecture and design with regard to the techniques/measures that are recommended by the Table E4 of the CENELEC Standard 50 129 for SIL 4 Sub-Systems
- Compliance of the ATC Trainborne Sub-System (EVC Platform) design characteristics with regard to the design features that are recommended by the Table E5 of the CENELEC Standard 50 129 for SIL 4 Sub-Systems;
- Lists of defensive design measures of the ATC Trainborne Sub-System (EVC Platform) that enable to prevent from creating an unacceptable risk in case of a Systematic Fault or of a Common Cause Failure
- Compliance of the ATC Trainborne Sub-System (EVC Platform) Safety Analyses with regard to the "Failure and Hazard Analysis Methods" that are recommended by the Table E6 of the CENELEC Standard 50 129 for SIL 4 Sub-Systems
- The documentary evidence, as well as the demonstration by "Verification and Testing" the functional, hardware and software requirements of the ATC Trainborne Sub-System are fulfilled, is provided in the "Safety Management Report"
- Compliance of the ATC Trainborne Sub-System software development process with regard to the SIL 4 requirements that are recommended by the CENELEC Standard 50 128
- Compliance of Verification and Validation techniques/measures used on the ATC Trainborne Sub-System with regard to the techniques/measures that are recommended by the Table E9 of the CENELEC Standard 50 129 for SIL 4 Sub-Systems.

5.3.4 Compliance Management

Compliance has to be managed at two levels: compliance editing and compliance reporting.

Compliance editing required the Eclipse client to be used to define the compliance maps between Baseline model elements and assurance assets elements. Figure 78 shows the GUI to specify Compliance Maps for artefact assets. For the Belgium GATC assurance project, the whole set of base artefacts have full compliance maps, since this project accomplished all the obligations in order to be certified.
Implementation of use cases on top of OPENCOSS platform

D1.4

Compliance reporting is done through the OPENCOSS Web client.

Compliance reports provide extensive functionality which helps OPENCOSS platform users to assess the current compliance of their project to the selected safety standard (i.e., baseline). Figure 79 shows an excerpt of the compliance report for Belgium GATC project artefacts required by the EN 50126 standard.

Gap Analysis reports facilitate the compliance assessment by viewing a Gap Analysis and viewing Evidence Evaluation results. Figure 80 shows an excerpt of a gap analysis report for Belgium GATC project artefacts required by the EN 50126 standard.

Finally, a metrics report is available in order to measure most of the indicators for our case study. Figure 81 provides an example of argumentation metrics for the Belgium GATC assurance project.
5.3.5 Cross-project reuse to create a new Assurance Project (France)
The goal now is to create a new assurance project for Alstom France by using the existing assurance project for Alstom Belgium, considering the product upgrade in terms of a new functionality: configurable braking curves.

Figure 82 shows the OPENCOSS user interface to reuse data from one project into another. This functionality allows users to reuse only the selected source evidence models associated to the active Assets Package, because evidences are not related to any other model of the project, or all the baselines associated to the active “Baseline Config” and all the evidence, argumentation and process models of the active Assets Package. All the models will be cloned to assure the integrity of the data, for example, a baseline could be related with argumentations, evidences and/or processes and in this way we are sure all the related information is copied avoiding inconsistencies.

Figure 82. Cross-project reuse from Belgium GATC to France GATC assurance project

Figure 83 shows the compliance report after reusing the Belgium GATC project into the France GATC project. There are a number of artefacts which must be completed and reviewed (compliance is partial) considering the new functionality of configurable braking curves.
5.3.6 Case Study Implementation from the external Process Tool "Process Director"

This section describes how the Railway Case Study has been implemented into the external tool "PTC Integrity Process Director" (formerly "Atego Process Director").

Process Director is tailored for supporting the CCL meta-model, then each capture of processes respects the CCL concept. Also the integration of Process Director with OPENCOSS provides features for importing process-related data as it expected by the OPENCOSS platform. Process-related data to be imported into OPENCOSS are fully consistent, that is to say compliant with the CCL meta-model. To ensure this compliance, the external Process Tool controls before importing the CCL conformance during the exportation step from Process Director (for example: relationship between CCL elements):

1. Each Use Case is defined as a Process Definition with all the relationships between reference Activities, between reference Activity and other CCL Elements: Artefact, Technique, and Role. If Process Definition is CCL compliant, it can be imported into OPENCOSS platform as a reference Framework.

2. From this Process Definition, a Project Definition is inherited and possibly tailored to match specific options. If Project definition is CCL compliant, it can be imported into OPENCOSS platform as a Project Model.

3. CCL meta-model requiring Participant relationship with Activity of Process Model, participants are assigned to their roles for each project Activity. Then the Project is executed: started with a start date, completed with an end date. At any time, if Project Progress is consistent, it can be imported for updating into OPENCOSS platform Project Execution.

5.3.6.1 Implementation of EN50126 from the external tool “Process Director” to the Opencoss tool
The figures below show how the railways EN50126 standard has been implemented into "Process Director", complying with the CCL meta-model.

**Figure 84 EN-50126 Library Structure in Process Director**

**Figure 85 Example of EN-50126 reference activity with its CCL attributes**
Implementation of use cases on top of OPENCOSS platform

Figure 86 Extract of EN-50126 BPMN Structure in Process Director

Figure 87 CCL conformance result before importing EN-50126 into OPENCOSS
5.3.6.2 Implementation of EN50128 from the external tool “Process Director” to the Opencoss tool

The figures below show how the railways EN50128 standard has been implemented into "Process Director", complying with the CCL meta-model.

Figure 88 Extract of EN-50128 Library Structure in Process Director
Implementation of use cases on top of OPENCOSS platform

Figure 89 Example of EN-50128 reference activity with its CCL attributes

Figure 90 Top of EN-50128 BPMN Structure in Process Director
5.3.6.3 Implementation of railways GATC France from the external tool “Process Director” to the Opencoss tool

The figures below show how the railways GATC France has been implemented into "Process Director", complying with the CCL meta-model.

Figure 91 CCL conformance result before importing EN-50128 into OPENCOSS
Implementation of use cases on top of OPENCOSS platform

Figure 92 GATC France Library Structure in Process Director

Figure 93 CCL conformance result before importing GATC France into OPENCOSS
5.1 General conclusion and improvement

The Railway case study relates to the certification of a GATC subsystem of the Alstom GATC solution for the ETCS of the ERTMS, known as EVC in the architecture of ERTMS. The GATC trainborne is an existing subsystem and already certified. The objective is not to re-certify the subsystem but to evaluate the benefits of the model-based certification approach on a real Alstom Transport subsystem. The case study will also deal with the reuse of the GATC subsystem from one country (Belgium) to another (France). Thus, the railway case study pays special attention to cross-country assurance/certification.

CCL has been validated on this case study showing its completeness for modelling the full set of assurance assets in Alstom projects. Most of the main “concerns” at the early CCL validation phases corresponded to the need for deciding how some aspects should be modelled to create meaningful representations of a standard or Alstom’s practices. This refers to, for example, the granularity that the artefacts, activities, etc., should have in the OPENCOSS platform.

It has been demonstrated that we can easily reuse assets from one assurance project into another and that we can easily identify the delta effort of product upgrades in terms of assurance.
6 Conclusions

Deliverable D1.4 summarizes the implementation of OPENCORES case studies. There are three case studies:

- Automotive Case Study deals with compositional certification based on the SEooC concept.
- Avionics Case Study focuses on cross-domain assurance and certification.
- Railway Case Study pays special attention to cross-country assurance/certification.

The OPENCORES consortium decided to follow an incremental approach for research and development, by releasing three prototype iterations. During the first prototype, it was created a CCL draft containing features for standards modelling, evidence characterization, process modelling, argumentation modelling, and compliance specification. In Prototype 2, CCL has been completed to cover cross-domain reuse by including the “equivalence map” concept. Also, the compositional assurance specification approach now allows tool users to specify assumptions, guarantees and contracts when integrating components into larger systems. In Prototype 3, the CCL Vocabulary approach has been developed as well as the mechanisms to validate “contracts” in the context of the compositional assurance approach. Also, the cross-domain and cross-project approaches were refined to allow users take reuse decisions in a more assisted way. Additionally, the link with the Atego Process Director tool has been conceptually refined to get a more precise translation of concepts to the OPENCORES platform.

The case study implementation evolved according to the following activities:

**Prototype 1.** Data from industrial partners covered different aspects of the assurance and certification process and in some cases have different levels of granularity, detail and completeness. Prototype 1 focused on evaluating the following aspects:

- Availability of industrial data to perform a meaningful excerpt of the case studies.
- Ability to extract the required information from case studies data so as to model them to achieve OPENCORESS goals (guidance, reuse, process automation, etc.).

**Prototype 2.** Industrial case studies centered on specific modeling aspects:

- The avionics case study studied a basic approach for cross domain reuse, which is based on the Equivalence Mapping concept. As a first step, we modelled the equivalence between CEN 50128 and DO-178C standards at the level of “compliance requirements” and “artefacts”
- The automotive case study focused on modelling SEooC assumptions, which is the basis for the contract-based approach developed by OPENCORESS.
- The railway case study focused on modelling the safety case for a subsystem, using the OPENCORESS argumentation editors.

**Prototype 3.** Industrial case studies centered on specific modeling aspects:

- The avionics case study extended the cross-domain reuse approach to cover the whole set of equivalence mappings between CEN 50128 and DO-178C standards, and used the concept of compliance post-conditions/obligations to complete the assistance for reuse decisions.
- The automotive case study focused on modelling the whole set of compositional concepts: assumptions, guarantees and contracts. The ePark subsystem was used as a SEooC integrated in a vehicle. This completes the compositional approach for assurance
- The railway case study focused on modelling the two assurance projects used to measure cross-country reuse efforts with OPENCORESS.