ECSEL Research and Innovation actions (RIA)

AMASS

Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems

Case studies description and business impact D1.1

Work Package:	WP1 Case Studies and Benchmarking
Dissemination level:	PU = Public
Status:	Final
Date:	11 May 2018
Responsible partner:	Bernhard Winkler (ViF)
Contact information:	bernhard.winkler@v2c2.at
Document reference:	AMASS_D1.1_WP1_VIF_V1.3

This deliverable is part of a project that has received funding from the ECSEL JU under grant agreement No 692474. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and from Spain, Czech Republic, Germany, Sweden, Italy, United Kingdom and France

Contributors

Names	Organisation
Bernhard Winkler, Helmut Martin	Virtual Vehicle (VIF)
Elena Alaña Salazar, Javier Herrero Martín	GMV Aerospace and Defence (GMV)
Carlo Vertua, Daniele Tornaghi	Thales Italia (THI)
Miguel Gómez, Juan Castillo	Thales Alenia Space España (TAS)
Anna Carlsson	OHB Sweden (OHB)
Mario Petrick, Mathias Killer, Behrang Monajemi	Assystem Germany (B&M)
Tomáš Kratochvíla, Vít Koksa	Honeywell International (HON)
Norbert Bartsch	Lange Research Aircraft (LAN)
Fredrik Warg, Martin Skoglund, Kenneth Östberg	Rise Research Institutes of Sweden (SPS)
Jan Hellberg, Alexander Åström	Comentor (COM)
David Deharbe	Clearsy (CLS)
Benito Caracuel	Scheneider Electric España (TLV)
Dian Nugraha, Farhan Bin Khalid, Frank Badstübner	Infineon Technologies (IFX)
Marc Born	Ansys Medini Technologies (KMT)
Garazi Juez, Huáscar Espinoza	Tecnalia Research & Innovation (TEC)
Fernando Mejia, Fabien Belmonte	Alstom Transport (ALS)

Reviewers

Names	Organisation
Mohamed Bakkali (Peer-reviewer)	Alliance pour les Technologies de l'Informatique (A4T)
Thomas Gruber (Peer-reviewer)	AIT Austrian Institute of Technology (AIT)
Petr Böhm (Peer-reviewer)	AIT Austrian Institute of Technology (AIT)
Cristina Martínez (Quality Manager)	Tecnalia Research & Innovation (TEC)
Garazi Juez	Tecnalia Research & Innovation (TEC)

Document History

Version	Date	Status	Author (Partner)	Remarks
V1.0	2016-11-30	First final version	C. Martinez (TEC)	
V1.1	2017-04-12	CS6 update	B. Winkler(ViF)	Alstom CS substitutes AVL CS&US.
V1.2	2018-02-09	Final version	B. Winkler(ViF)	First review comments addressed (IFX CS2, ALS CS6 and HON CS7).
V1.3	2018-05-11	Final version	C. Martinez (TEC)	First review comments addressed (IFX CS2, ALS CS6 and HON CS7).

TABLE OF CONTENTS

Executive Summary........7
1. Introduction........8
2. Case Study Description........10
2.1 CS1: Industrial and Automation Control Systems (IACS)........10
2.1.1 Short description of the case study........10
2.1.2 Technical description of the case study........11
2.1.3 Case study state of the art........13
2.1.4 Case study state of the practice........14
2.1.5 Expected technical improvements........17
2.1.6 Business needs........18
2.1.7 Usage scenarios........19
2.2 CS2: Advanced driver assistance function with electric vehicle........22
2.2.1 Short description of the case study........22
2.2.2 Technical description of the case study........22
2.2.3 Case study state of the art........23
2.2.4 Case study state of the practice........23
2.2.5 Expected technical improvements........25
2.2.6 Business needs........26
2.2.7 Usage scenarios........27
2.3 CS3: Collaborative automated fleet of vehicles........29
2.3.1 Short description of the case study........29
2.3.2 Technical description of the case study........29
2.3.3 Case study state of the art........32
2.3.4 Case study state of the practice........32
2.3.5 Expected technical improvements........34
2.3.6 Business needs........35
2.3.7 Usage scenarios........36
2.4 CS4: Design and safety assessment of on-board software applications in Space Systems........40
2.4.1 Short description of the case study........40
2.4.2 Technical description of the case study........41
2.4.3 Case study state of the art........42
2.4.4 Case study state of the practice........42
2.4.5 Expected technical improvements........45
2.4.6 Business needs........46
2.4.7 Usage scenarios........47
2.5 CS5: Platform Screen Doors Controller........49
2.5.1 Short description of the case study........49
2.5.2 Technical description of the case study........49
2.5.3 Case study state of the art........52
2.5.4 Case study state of the practice........52
2.5.5 Expected technical improvements........54
2.5.6 Business needs........54
2.5.7 Usage scenarios........55
2.6 CS6: Automatic Train Control Formal Verification........57
2.6.1 Short description of the case study........57
2.6.2 Technical description of the case study........57
2.6.3 Case study state of the art........58

2.6.4 Case study state of the practice 59
2.6.5 Expected technical improvements 62
2.6.6 Business needs 63
2.6.7 Usage scenarios..........65
2.7 CS7: Safety Assessment of Multi-Modal Interactions in Cockpits 66
2.7.1 Short Description of the Case Study 66
2.7.2 Technical Description of the Case Study..........66
2.7.3 Case Study State of the Art 67
2.7.4 Case study state of the practice 68
2.7.5 Expected technical improvements 70
2.7.6 Business needs 71
2.7.7 Usage scenarios..........71
2.8 CS8: Telematics Function 75
2.8.1 Short description of the case study 75
2.8.2 Technical description of the case study..........75
2.8.3 Case study state of the art 77
2.8.4 Case study state of the practice 77
2.8.5 Expected technical improvements 79
2.8.6 Business needs 79
2.8.7 Usage scenarios..........80
2.9 CS9: Safety-Critical SW Lifecycle of a Monitoring System for NavAid (ATM domain) 81
2.9.1 Short description of the case study 81
2.9.2 Technical description of the case study..........81
2.9.3 Case study state of the art 82
2.9.4 Case study state of the practice 83
2.9.5 Expected technical improvements 84
2.9.6 Business needs 85
2.9.7 Usage scenarios..........86
2.10 CS10: Certification basis to boost the usage of MPSoC architectures 88
2.10.1 Short description of the case study 88
2.10.2 Technical description of the case study..........88
2.10.3 Case study state of the art 88
2.10.4 Case study state of the Practice 89
2.10.5 Expected technical improvements 91
2.10.6 Business needs 91
2.10.7 Usage scenarios..........93
2.11 CS11: Design and efficiency assessment of model based Attitude and Orbit Control software development 95
2.11.1 Short description of the case study 95
2.11.2 Technical description of the case study..........95
2.11.3 Case study state of the art 96
2.11.4 Case study state of the practice 97
2.11.5 Expected technical improvements 98
2.11.6 Business needs 99
2.11.7 Usage scenarios..........100
3. Questionnaire Evaluation 103
4. Conclusions..........106
Abbreviations and Definitions..........107
References 110

List of Figures

Figure 1. Number of case studies per AMASS domain 8
Figure 2. Smart Grid: Schneider Electric vision 10
Figure 3. Schneider Electric Saitel® RTU platform 11
Figure 4. Electrical substation and IACS....11
Figure 5. Control Center....12
Figure 6. Experimental scenario architecture 13
Figure 7. Product design scope 16
Figure 8. Safety and Security Workflow of the SESAMO project 24
Figure 9. Cooperative Adaptive Cruise Control (CACC)....29
Figure 10. Platoon topologies 30
Figure 11. State Machine Diagram for graduation and degradation of assistant systems....31
Figure 12. Functional architecture of CACC 32
Figure 13. Iberian Peninsula [9]....40
Figure 14. OLCI instrument 41
Figure 15. Development life-cycle....44
Figure 16. Example of combined systems to ensure safety at a metro platform 49
Figure 17. Platform screen doors COPPILOT....50
Figure 18. COPPILOT SP architecture....51
Figure 19. COPPILOT SP languages 53
Figure 20. Alstom's CBTC development life-cycle 59
Figure 21. CS6 Workflow....60
Figure 22. EN 50129 organization 62
Figure 23. Multifunction controls....66
Figure 24. Logical view of telematics functions for a connected vehicle. 75
Figure 25. Positioning function for Core iteration.....76
Figure 26. Positioning function and ITS facility for P1/P2 iterations. 77
Figure 27. V-model for safety and security....78
Figure 28. CS9 workflow 83
Figure 29. Electra, Telecommunication satellite 95
Figure 30. AOCS SW Development Cycle 96
Figure 31. Relevant domains for the AMASS partners 103
Figure 32. Professional categories of AMASS partners....103
Figure 33. Overview of number employees of AMASS partners....103
Figure 34. Partners activities in the development process....104
Figure 35. Partners activities in the assurance process....104
Figure 36. Partners activities in the certification process....105
Figure 37. Scientific and Technical Objectives (STOs) covered by AMASS case studies....106
Figure 38. AMASS goals covered by case studies....106

List of Tables

Table 1. AMASS Case studies 9
Table 2. Impact of IEC 61508 Standards on Intelligent Electrical Networks and Safety Improvement [15] 14
Table 3. Tool chains used in the RTU design process 17
Table 4. CS1 TLV and TEC usage scenarios 19
Table 5. CS1 TEC, KMT and AIT usage scenarios 20
Table 6. CS1 FBK and A4T usage scenarios 21
Table 7. CS2 IFX usage scenarios 27
Table 8. CS2 TEC and KMT usage scenarios 28
Table 9. CS3 Workflow 33
Table 10. CS3 B&M usage scenarios 36
Table 11. CS3 B&M and TEC usage scenarios 37
Table 12. CS3 KMT, MDH, AIT and ViF usage scenarios 38
Table 13. CS3 ViF and A4T usage scenarios 39
Table 14. State of practice of Sentinel-3 OEU ICM SW 43
Table 15. CS4 GMV and TEC usage scenarios 47
Table 16. CS4 TEC, FBK, INT and TAS-E usage scenarios 48
Table 17. CS5 CLS and AIT usage scenarios 55
Table 18. CS5 FBK, A4T and CEA usage scenarios 56
Table 19. CS6 ALS usage scenarios 65
Table 20. CS7 Workflow – baseline – tools used for development before AMASS 68
Table 21. CS7 Workflow – expected tools to be used for development thanks to AMASS 68
Table 22. Tools used in Case Study 7 69
Table 23. Proposed tool chain to be deployed for Case Study 7 70
Table 24. CS7 HON and TEC usage scenarios 72
Table 25. CS7 TEC, FBK and INT usage scenarios 73
Table 26. CS7 UOM and LAN usage scenarios 74
Table 27. CS8 SPS and COM usage scenarios 80
Table 28. CS9 THI usage scenarios 86
Table 29. CS9 THI and INT usage scenarios 87
Table 30. CS10 TAS-E and TEC usage scenarios 93
Table 31. CS10 INT and RPT usage scenarios 94
Table 32. CS11 Workflow 97
Table 33. CS11 Tool Chain 98
Table 34. CS11 OHB and MDH usage scenarios 100
Table 35. CS11 MDH, FBK and INT usage scenarios 101
Table 36. CS11 RPT usage scenarios 102

Executive Summary

The deliverable D1.1 “Case studies description and business impact” is released by the AMASS work package WP1 “Case Studies and Benchmarking” and describes the 11 industrial case studies of the AMASS project.

For each case study the deliverable provides:

- Short description (overview) about the case study.
- Detailed technical description of the case study.
- Description of the state of the art regarding the case study.
- Description of the state of the practice in the partner companies regarding the case study.
- A rough description of the assurance workflow in the case study.
- Description about the assessment work by a third party or any independent company internal department.
- Which roles are involved in the case study and how many people work in it.
- Which methods and tools are used and how they interoperate.
- The expected improvement regarding the case study at the end of the AMASS project and which Scientific and Technical Objectives (STOs) are covered.
- Description of the specific business needs to improve the case study and which AMASS goals are covered.
- Definition of the case study stakeholders and the practices developed by them.

This deliverable also provides the main results of the questionnaire which was performed to get more information about the AMASS project partners and to obtain independent information. The questionnaire evaluation shows the main partners’ activities in the development process, the assurance and certification process. A further result of the questionnaire is an overview about the partners’ relevant domains, the professional categories and the number of employees.

D1.1 relates to the following AMASS deliverables:

- D1.2 “Report of case study data collection” [m12], which will present the data collected for the execution of each case study.
- D2.1 “Business cases and high-level requirements” [m11], which content will be partially derived from the description of the case studies.

1. Introduction

AMASS will create and consolidate a de-facto European-wide assurance and certification open tool platform, ecosystem and self-sustainable community spanning the largest CPS vertical markets. The ultimate aim is to lower certification costs in face of rapidly changing product features and market needs. This will be achieved by establishing a novel holistic and reuse-oriented approach for architecture-driven assurance (fully compatible with standards such as AUTOSAR and IMA), multi-concern assurance (compliance demonstration, impact analyses, and compositional assurance of security and safety aspects), and for seamless interoperability between assurance/certification and engineering activities along with third-party activities (external assessments, supplier assurance).

This report presents the results of task 1.1 Case Study Specification and will define and elaborate the industrial case studies that correspond to the scope addressed by AMASS.

The aim is to specify the industrial case studies as required for covering the different aspects of the project that are needed to demonstrate and evaluate the AMASS platform improvements.

The case studies provide a set of AMASS user needs. Those user needs are derived from the industrial application domains, the AMASS industrial partners, and best practices applied on the assurance and certification/qualification of safety/security-critical products. The AMASS user needs are identified through questionnaires and interviews to industrial partners.

The task is based on the industrial practices employed by the different stakeholders, which are related to the “re-use” approach in the development, assurance and certification process. Furthermore, D1.1 presents the impact on the different business cases, which will result from the project: constraints on methodology, constraints on standards, and constraints on industry processes or practices are to be updated according to the results. This is needed to guarantee the coverage and completeness of industrial case studies.

In Section 2, the 11 AMASS case studies and the case study related usage scenarios are described.

The 11 AMASS case studies cover the following different domains (see Figure 1):

- 3 Automotive domain
- 3 Space domain
- 2 Railway domain
- 1 Avionics domain
- 1 Air Traffic Management (ATM) domain
- 1 Industrial Automation domain

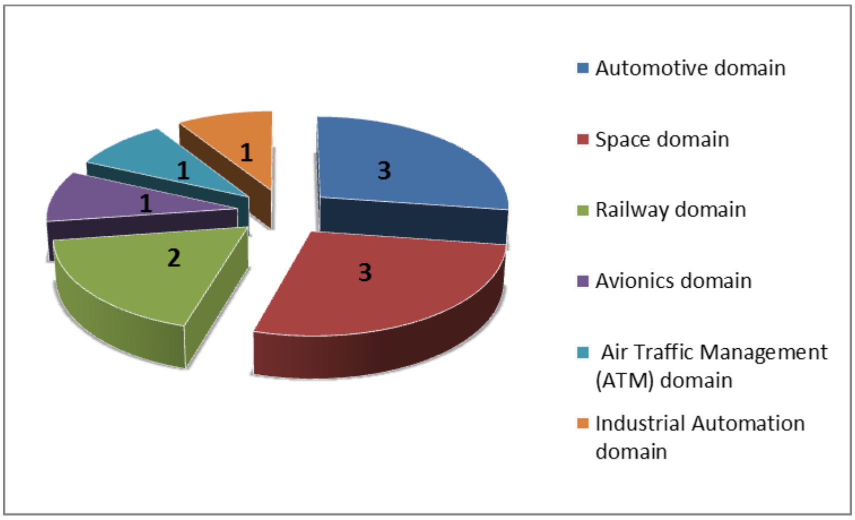


Figure 1. Number of case studies per AMASS domain

The 11 case studies are listed in Table 1.

Table 1. AMASS Case studies

CSNr	Partner	Short	Case Study	Domain
CS1	Schneider Electric España S.A.	TLV	Industrial and Automation Control System	Industrial Automation domain
CS2	Infineon	IFX	Advanced driver assistance function with electric vehicle	Automotive domain
CS3	Berner & Mattner	B&M	Collaborative automated fleet of vehicles	Automotive domain
CS4	GMV Aerospace and Defence, S.A.U.	GMV	Design and safety assessment of on-board software	Space domain
CS5	CLEARSY SAS	CLS	Platform Screen Doors Controller	Railway domain
CS6	Alstom Transport SA	ALS	Automatic Train Control Formal Verification	Railway domain
CS7	Honeywell	HON	Safety assessment of multi-modal interactions in cockpits	Avionics domain
CS8	SP Sveriges Tekniska Forskningsinstitut	SPS	Telematics function	Automotive domain
CS9	Thales Italia SpA	THI	Safety-Critical SW Lifecycle of a Monitoring	Air Traffic Management domain
CS10	Thales Alenia Space	TAS-E	Certification basis to boost the usage of MPSoC architectures	Space domain
CS11	OHB Sweden AB	OHB	Design and efficiency assessment of model-based Attitude	Space domain

Section 3 provides a summary and evaluation of the questionnaire. The main goal of the questionnaire is to collect partner specific data (not case study specific data) which could be used by another AMASS WPs.

The questionnaire includes general questions about the partner and the partner activities in the development process, the assurance process, and the certification process.

Section 4 concludes the deliverable and provides a summary and an overview of the elaborated results.

2. Case Study Description

2.1 CS1: Industrial and Automation Control Systems (IACS)

2.1.1 Short description of the case study

The European Commission defines Critical Infrastructure as “asset or system which is essential for the maintenance of vital societal functions. The damage to a critical infrastructure, its destruction or disruption by natural disasters, terrorism, criminal activity or malicious behaviour, may have a significant negative impact for the security of the EU and the well-being of its citizens”. For European Union, the protection of these infrastructures is one of the major objectives. An adequate level of protection must be ensured and the detrimental effects of disruptions on the society and citizens must be limited as far as possible.

The European Programme for Critical Infrastructure Protection (EPCIP) defines the **Smart Grid** as critical infrastructure. The adoption of smart grids is transforming the traditional energy grids, bringing control systems and devices with new requirements on the control, monitoring and protection of distribution substations and transformer stations/centres.

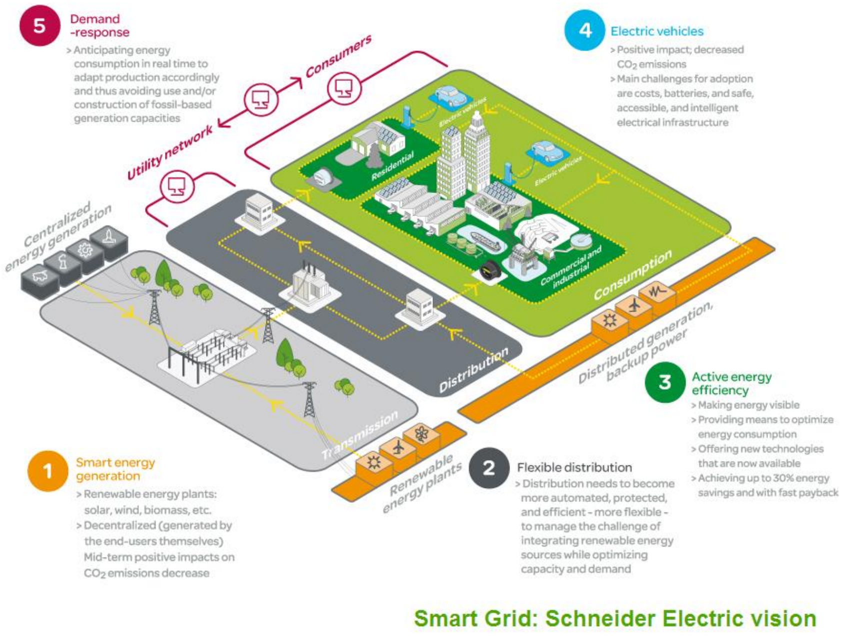

5
Demand -response
> Anticipating energy consumption in real time to adapt production accordingly and thus avoiding use and/or construction of fossil-based generation capacities
Consumers
Utility network
4
Electric vehicles
> Positive impact; decreased CO_2 emissions
> Main challenges for adoption are costs, batteries, and safe, accessible, and intelligent electrical infrastructure
Electric vehicles
Residential
Electric vehicles
Centralized energy generation
Commercial and industrial
Consumption
Distributed generation, backup power
Distribution
Transmission
3
Active energy efficiency
> Making energy visible
> Providing means to optimize energy consumption
> Offering new technologies that are now available
> Achieving up to 30% energy savings and with fast payback
Renewable energy plants
1
Smart energy generation
> Renewable energy plants: solar, wind, biomass, etc.
> Decentralized (generated by the end-users themselves) Mid-term positive impacts on CO_2 emissions decrease
2
Flexible distribution
> Distribution needs to become more automated, protected, and efficient - more flexible - to manage the challenge of integrating renewable energy sources while optimizing capacity and demand
Smart Grid: Schneider Electric vision

Figure 2. Smart Grid: Schneider Electric vision

One of the key elements of the Smart Grid is the **Industrial and Automation Control Systems (IACS)**. The IACS control and monitor the electrical infrastructure. These control systems are composed by advanced embedded systems named **Remote Terminal Units (RTU)** that are evolving to become smart devices, and enclose serial and Ethernet communications, data logging capabilities, analog and digital inputs/outputs, etc.

Figure 3. Schneider Electric Saitel® RTU platform

The Smart Grid implies advanced functionalities with the involvement of new actors. New safety and security requirements must be taken into account in this scenario, standing out standards, such as: **IEC 61508 and IEC 62351**. In this sense, the challenge now is to **increase the safety and security** aspects of the embedded systems that manage the electrical network.

2.1.2 Technical description of the case study

The case study CS1 focuses on the smart grid domain, in particular it is based on an Industrial and Automation Control Systems of the electrical substation. The substation is composed by several devices such as: sensors and actuators, remote terminal units, programmable logic controllers, supervisory control and data acquisition (SCADA), human interface machine (HMI), etc. The IACS is the element that monitors and controls the substation and sends/receives information to/from the control center.

Figure 4. Electrical substation and IACS

Figure 5. Control Center

Schneider Electric, as IACS industry global leader, will provide the suitable experimental scenario that will be used in the project activities under realistic conditions. The aim will be to emulate a real IACS of the Electrical Distribution Network. Therefore, the scenario will cover from the highest entity in the control hierarchy, the SCADA system, to the field elements, the RTUs. Moreover, it will include the three main levels of typical smart grid control system architecture:

- **Level 1 – Field Site**: a subsystem that houses the acquisition system. This level runs acquisition and control activities to gather data and send it to the control center. Common devices present in the field site are the RTUs. These devices are critical assets equipped with input and output signals that provide control, monitoring and data gathering functions to the control substation, which is part of the Smart Grid. The scenario will include RTUs with acquisition signals, Ethernet and serial communications, and logical programming (Schneider Electric Saitel® RTUs).
- **Level 2 – Front End**: This level is in charge of concentrating the communications between the acquisition system (level 1) and the SCADA (level 3). Therefore, the second level will include the elements needed for the communication with the field site and the control center. The scenario will include RTUs with several communication ports (Schneider Electric Saitel® RTUs).
- **Level 3 – Control Center**: this is the highest level of the control system and includes the SCADA system. The SCADA sends the commands to the acquisition system and receives information from the field site (Schneider Electric OASyS® SCADA).

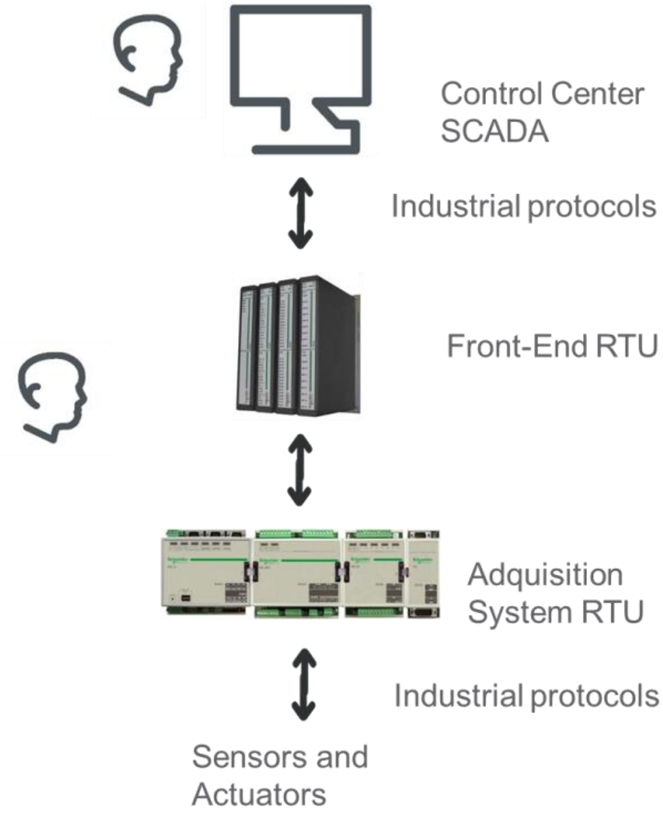


Figure 6. Experimental scenario architecture

In conclusion, the case study CS1 will cover the functionality and equipment of a typical IACS. It will be based in a master-slave configuration, composed by several RTUs (control, communications, and acquisition), including the industrial protocols that are presented in the electrical networks, such as: DNP 3, Modbus and IEC 60870-5-104.

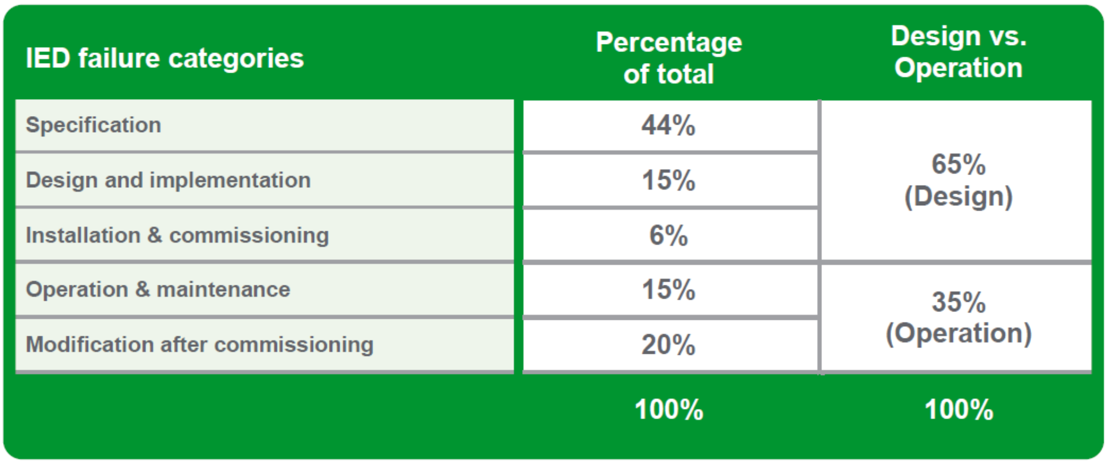
2.1.3 Case study state of the art

Over the last several years, utilities have replaced electro-mechanical technologies with new programmable electronic systems. While utilities have benefitted from the new technologies, it is difficult for operations personnel to determine every possible failure scenario and to predict issue-related network behaviours. The stakes are high as the tolerance for medium / high voltage electrical network downtime continues to erode. Costs are too high for both customers and utilities when network failures occur. In addition, the need to maintain safe network operation is a growing concern given the increase in complexity of the emerging networks.

These programmable electronic systems (also referred to as Intelligent Electronic Devices or IEDs), are characterized by failure modes that are different from the traditional electro-mechanical relays. The IEDs contain hundreds of electronic components and have software embedded into their microprocessors. This results in increased network complexity.

The risks are real. According to a study conducted by the UK Health and Safety Executive, 65% of incidents involving process control systems occur during the specification, design, installation and commissioning phases of the product implementation [20]. The rest occur during the maintenance and modification that take place after commissioning (see next table).

Table 2. Impact of IEC 61508 Standards on Intelligent Electrical Networks and Safety Improvement [15]



IED failure categories	Percentage of total	Design vs. Operation
Specification	44%	65% (Design)
Design and implementation	15%	
Installation & commissioning	6%	
Operation & maintenance	15%	35% (Operation)
Modification after commissioning	20%	
	100%	100%

For effective management of IED devices, risk reduction can be best achieved through the execution of robust design principles. Fortunately, **industry standards such as IEC 61508** have been introduced that provide guidance on how to improve modern electrical network safety performance.

The IEC 61508 standard defines a methodology for engineering safety functions that allows all the relevant factors, associated with a product or application, to be fully taken into account and thereby meet the specific needs of users of the product and the application sector. This standard is widely used by electronic device manufacturers and suppliers when any part of the safety function contains an electrical, electronic, or programmable electronic component and where application sector international standards do not exist.

The IEC 61508 standard specifies the risk assessment and the measures to be taken in the design of safety functions for the avoidance and control of faults. In fact, IEC 61508 provides a complete safety life cycle that accounts for possible risk of physical injury and damage to the environment.

Cybersecurity is another concern in the industrial automation domain. In the last years, the systems of this domain are exposed to cyber-attacks. In this context, the standard IEC 62351 “Information Security for Power System Control Operations” is the main reference for cyber security in the electrical substation. This standard covers the cyber security of the electrical infrastructure in several aspects: access control, communications and protocols, even register, and others.

2.1.4 Case study state of the practice

Currently, the Saitel® RTU devices are checked against requirements and specifications with the following Test Plans:

1. **Product and System Verification Plan.** Provides a test plan for product & system that shows the general test methods for designing functional verification and for product & system verification. It covers two parts:
    a. **Design functional verification plan.** These tests are performed during the product design & functional verification phase, when no industrial product yet exists. For this reason, tests are performed on “models”, theoretical or experimental, simulating the product. These tests are intended to assure that the design, as described in the product design files, meets its requirements as described in the product specification.

    b. **Product & system verification tests.** These tests are performed on prototypes issued from manufacturing, during implementation & qualification phase. They are conducted before the validation tests, on simulated environment, in order to check performances, functional limits, and the margin of the product. For this reason, tests can be performed on prepared products, in order to provide appropriate estimation.

The verification plan includes some safety tests (Failure Modes and Effects Analyses (FMEA) for digital outputs modules).

2. **Product and System Validation Plan.** This plan describes the overall product test strategy that organizes and optimizes all tests to be performed on the product. The tests to evaluate the equipment are of different nature:
   - **Functional Tests**: to check the communications, inputs and output signals, diagnostics and signalling, firmware upgrading, etc.
   - **Electrical Safety Tests**: Isolation Resistance Measurement and Dielectric Rigidity.
   - **Electrical Tests**: consumption, protections, tolerances, etc.
   - **Ambient Tests**: are focused on safety in certain environments and can also be considered as life-tests. They are based on standard IEC 60068-2.
   - **Electromagnetic Compatibility (EMC) Tests**: Electrostatic Discharges, Electric Fast Transients (Burst), Surge. According to UNE–EN 610006-4 (Emissions) and UNE–EN 610006-2 (Immunity).
   - **Reliability Test**: Mean Time Between Failures (MTBF) test is carried out according to the MIL-HDBK-217-F (Military Handbook: Reliability Prediction of Electronic Equipment).
   - **Cyber security Tests**: access control, communications and event register according to IEC 62351.
   - **Mechanical Tests**: enclosure, connectors, identification, etc.

In addition, before the production phase, the devices are tested by an external certification authority to obtain the EMC certifications required by the customers.

Finally, after the production phase, each manufactured module must be checked with the **Module Inspection Test (MIT) procedure**. The purpose of the MIT is verifying that the functionality of the manufactured module complies with specific requirements.

2.1.4.1 Workflow

The following figure shows the standard process to design and test new RTU products and product evolutions:

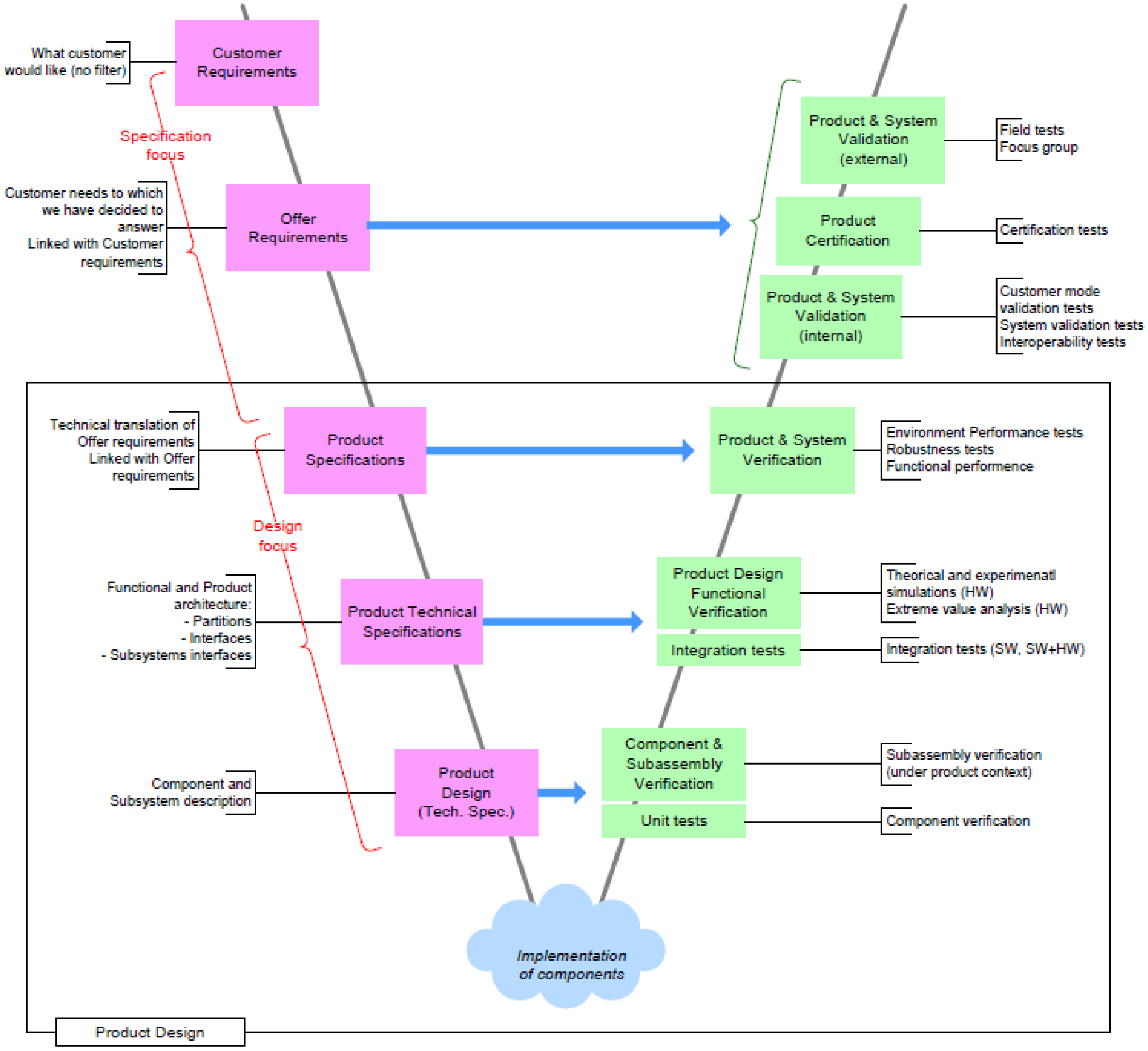

What customer would like (no filter)
Customer Requirements
Specification focus
Customer needs to which we have decided to answer
Linked with Customer requirements
Offer Requirements
Technical translation of Offer requirements
Linked with Offer requirements
Product Specifications
Design focus
Functional and Product architecture:
- Partitions
- Interfaces
- Subsystems interfaces
Product Technical Specifications
Component and Subsystem description
Product Design (Tech. Spec.)
Implementation of components
Product & System Validation (external)
Field tests
Focus group
Product Certification
Certification tests
Product & System Validation (internal)
Customer mode validation tests
System validation tests
Interoperability tests
Product & System Verification
Environment Performance tests
Robustness tests
Functional performence
Product Design Functional Verification
Theorical and experimenatl simulations (HW)
Extreme value analysis (HW)
Integration tests
Integration tests (SW, SW+HW)
Component & Subassembly Verification
Subassembly verification (under product context)
Unit tests
Component verification
Product Design

Figure 7. Product design scope

The safety and security aspects are considered in the design, verification and validation phases.

2.1.4.2 Assessment

There are four different assessments for RTU products:

- Product & System Verification plan (internal department)
- Product & System Validation plan (internal department)
- Product Certification (external)
- Product & System Validation (external)

2.1.4.3 Involved roles

In general, there are several roles involved in the assessment processes:

- Electronic Hardware Engineers and Electronic Software Engineers in the verification processes
- Electronic Test Engineers in the validation process

- Electronic Test Engineers (from external laboratory) in the certification process
- Electronic Test Engineers (from manufacturer) in post-production process

2.1.4.4 Tools and Tool chains

2.1.4.4.1 Used tools and methods (included guidelines)

Currently, there is not a tool to run the safety and security assessment for the RTUs. The target is to obtain a new tool that includes these two concerns and try to automate as much as possible the process. This tool shall consider the following standards as a reference:

- IEC 61508 (safety)
- IEC 62351 (security)

2.1.4.4.2 Tool chains

The following tool chains are used in the RTU design and development process:

Table 3. Tool chains used in the RTU design process

Hardware Development	Software Development
• HW scheme design: Orcad CIS (Cadence), HDL (Cadence) • PCB design: Allegro (Cadence) • Circuits simulation: Tina (Texas Instrument) • Functional testing: Visual Basic .Net (Microsoft) • Firmware development and testing: AVR Studio (Atmel), CodeWarrior (Freescale), STM Studio (ST), Coocox. • Firmware CPLD: Quartus (Altera) • BBDD of components: MySQL	• SW design: Eclipse, Visual Studio (Microsoft), Work bench 3.1 (Wind River) • SW testing: Scripts, Microsoft Excel

2.1.5 Expected technical improvements

In general, we expect to integrate new safety and security methodologies and tools to the RTU, based on the standards, such as: IEC 61508 and IEC 62351. The new AMASS tool would be integrated in the design and development RTU processes, including safety and security requirements in the workflow, improving the verification and validation and enabling the certification in these two aspects.

2.1.5.1 STO1. Architecture-driven Assurance (High Priority)

- SIL (Safety Integrity Level) estimation according to current practices.
- Link with safety/security analysis tools (impact on assurance & certification).
- Introducing safety and security concerns in the early phases of product development in order to reduce costs.
- Model-driven approaches facilitate component reuse.

2.1.5.2 STO2. Multi-concern Assurance (High Priority)

- Safety/security co-assessment (this includes several aspects such as safety and security co-design, co-analysis and co-V&V).
- Compliance management with IEC 61508 (Functional Safety) and IEC 62351 (Cybersecurity).

2.1.5.3 STO3. Seamless Interoperability (Medium Priority)

- Integration with current Schneider tool chain for RTU design & development.
- Integration with new safety/security analysis tools.

2.1.5.4 STO4. Cross/Intra-Domain Reuse (Low Priority)

- Reuse for product upgrades and for product families.

2.1.6 Business needs

2.1.6.1 AMASS Goal 1

G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.

- Effort for managing compliance with targeted standards.
- Effort to run safety/security analyses.

2.1.6.2 AMASS Goal 2

G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.

- Costs for assurance and re-certification of product upgrades or similar RTU developments.

2.1.6.3 AMASS Goal 3

G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.

- Help to estimate the costs/effort of future developments so that Schneider Electric reduces the risks of new developments/certifications.

2.1.6.4 AMASS Goal 4

G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.

- Reduce efforts to exchange data between tools (any tool that must interact with assurance and certification activities).

2.1.7 Usage scenarios

Table 4, Table 5 and Table 6 show the 10 usage scenarios related to Case Study 1.

Table 4. CS1 TLV and TEC usage scenarios

ID:	TLV UsageScenario 1	TLV UsageScenario 2	TLV UsageScenario 3	TEC UsageScenario 1
Related CaseStudy	CS1	CS1	CS1	CS1
Addressed Domains	Industrial Automation	Industrial Automation	Industrial Automation	Industrial Automation
Scenario Name	Assurance/Certification Management Tool	Model-Based Development for Safety and Security co-assessment	Reuse of components	Assurance/Certification Management Tool
Short Description	Compliance with Standards/ Product and process assurance/certification management tool to support the compliance assessment and certification	1) Support for Model-based System, Safety, and Security Co-Engineering 2) Support for Safety and Security Co-Analysis 3) Support for Safety and Security V&V 4) Architectural patterns: trade-off based on analysis and certification requirements 5) Fault Injection	Reuse of components from one system to another	Compliance with Standards/ product and process assurance/certification management tool to support the compliance assessment and certification
Stakeholders	Safety/Security Manager Assurance Manager Quality Manager Safety Assessor	Safety Engineer Security Engineer System Engineer	System engineer Safety engineer Security engineer Quality Assurance Manager	Safety/Security Manager Assurance Manager Quality Manager Safety Assessor
Stakeholder constraints	None	None	None	None
Addressed Business Goals:	G4	G4, G1, G3	G2	G4
Process Steps	1.- Establish Assurance & Compliance Objectives 2.- Prepare Evidences 3.- Get Ready the Certification Dossier 4.- Reuse in Future Dossiers	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification	All	1.- Establish Assurance & Compliance Objectives 2.- Prepare Evidences 3.- Get Ready the Certification Dossier 4.- Reuse in Future Dossiers
Concerns	Safety and Security Reliability	Safety and Security Reliability	Safety and Security Reliability	Safety and Security Reliability
Cross-system certification	Yes	Yes	Yes	Yes
Cross-domain certification	No	No	No	No
Engineering Environment (Interoperability)	OpenCert/ AMASS Platform	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools-Safety/Security Analyses Tool and V&V Tools (e.g. Fault Injection)	Open Source tools AMASS tools when available for use and evaluation	OpenCert/ AMASS Platform
Challenges	Safety and Security co-assessment	Safety and Security co-assessment	Reuse of components targeting items of different SILs (&security levels)	Safety and Security co-assessment
Standards	IEC 61508 IEC 62351	IEC 61508 IEC 62351	IEC 61508, IEC 62351	IEC 61508 IEC 62351
Any wishes for usage scenario	N/A	Reuse of established safety methods for security topic	Reuse of established safety methods for security topic	N/A
Any known constraints for usage scenario	Not so far	Not so far	Not so far	Not so far

Table 5. CS1 TEC, KMT and AIT usage scenarios

ID:	TEC UsageScenario 2	TEC UsageScenario 3	KMT UsageScenario 1	AIT UsageScenario 1
Related CaseStudy	CS1	CS1	CS 1	CS1
Addressed Domains	Industrial Automation	Industrial Automation	Industrial Automation	Industrial automation and Control
Scenario Name	Model-Based Development for Safety and Security co-assessment	Reuse of components		Security analysis & testing
Short Description	1) Support for Model-based System, Safety, and Security Co-Engineering 2) Support for Safety and Security Co-Analysis 3) Support for Safety and Security V&V 4) Architectural patterns: trade-off based on analysis and certification requirements 5) Fault Injection	Reuse of components from one system to another	Modeling of the system architecture with SysML and adding of relevant properties for safety (and security) to SysML components. Perform the FMEA and FTA based on these components. Demonstrate Tool-Flexibility by switching between IEC 61508 FMEDA and automotive HW Metrics from the same data. Demonstrate seamless interoperability with system design and requirement tools.	Efficient security and safety analysis, including threat modeling and security testing for security assurance for components
Stakeholders	Safety Engineer Security Engineer System Engineer	System engineer Safety engineer Security engineer Quality Assurance Manager	Safety Manager Requirments Engineer System engineer Safety engineer	Risk analyzer Safety engineer Security engineer Test engineer
Stakeholder constraints	None	None	None	None
Addressed Business Goals:	G4, G1, G3	G2	G1, G4	G1, G2
Process Steps	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification	All	Product development on system/HW level concerning safety (security) -System/HW requirements -System/HW design -System/HW analysis -System/HW modelling	Development and assurance process
Concerns	Safety and Security Reliability	Safety and Security Reliability	Safety (Security)	Security and safety
Cross-system certification	Yes	Yes	none	Produce assurance evidence supporting generic, component level security and safety case
Cross-domain certification	No	No	none	No
Engineering Environment (Interoperability)	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools-Safety/Security Analyses Tool and V&V Tools (e.g. Fault Injection)	Open Source tools AMASS tools when available for use and evaluation	medini analyze, Rhapsody or EA, any requirement management tool	Threat modeling tool, tool for security testing in test phase
Challenges	Safety and Security co-assessment	Reuse of components targeting items of different SILs (&security levels)	tool adaptation for application in another domain.	Development of a safety and security element out of context
Standards	IEC 61508 IEC 62351	IEC 61508, IEC 62351	IEC 61508	IEC 61508, IEC 62351, IEC 62443
Any wishes for usage scenario	Reuse of established safety methods for security topic	Reuse of established safety methods for security topic	System design done with SysML. Case study owner has to provide HW information and proper design models to support safety analysis	Availability of system specification and prototype systems for security analysis and testing
Any known constraints for usage scenario	Not so far	Not so far	none	generic system architecture makes it difficult to identify safety and security requirements and requires a generic approach which is validated as soon as the real usage scenario is known

Table 6. CS1 FBK and A4T usage scenarios

ID:	FBK UsageScenario CS1	A4T UsageScenario 1
Related CaseStudy	CS1	CS1
Addressed Domains	Industrial Automation	Industry
Scenario Name	CS1FBK	Security analysis & testing
Short Description	1) Modeling of the system architecture including the plant and the data acquisition devices 2) Formalization of the system requirements including functional, safety, security, and reliability requirements 3) Formalization of component requirements including requirements of monitoring components 4) Validation of the requirements including analysis of the diagnosability of the plant 5) Contract-based verification of requirements refinement 6) Modeling of the components behavior	Model Based Safety Analysis (MBSA) with safety demonstration
Stakeholders	System engineer, Safety & Security engineer, ModelBased Safety researcher, Verification & validation researcher	System engineer Safety engineer Security engineer
Stakeholder constraints	None	None
Addressed Business Goals:	G3	G1 - G2 - G3
Process Steps	system requirements system design system analysis system modeling system verification evidence for system argumentation	Product development on system level concerning safety/security -System requirements -System design -System argumentation
Concerns	Safety Security Reliability	Safety and Security
Cross-system certification		Between system components
Cross-domain certification	No	No
Engineering Environment (Interoperability)	CHESS interacting with analysis tools (OCRA, nuXmv, xSAP) and with OpenCert: Modeling in SySML or AADL using CHESS Formalization using CHESS/OCRA integration Validation and Refinement checked with OCRA	MBSE approach + Safety Architect
Challenges	Assurance of monitoring components Application of formal methods Generation of evidence Safety and security co-engineering Formalization and refinement of	Safety and Security co engineering - collaborative work between system architect and safety and security analysts
Standards	N/A	CEI 61508
Any wishes for usage scenario	N/A	N/A
Any known constraints for usage scenario	N/A	N/A

2.2 CS2: Advanced driver assistance function with electric vehicle

2.2.1 Short description of the case study

During the life of the AMASS project a complex automotive case study will be defined, which covers aspects down from vehicle level (where hazards and reliability requirements emerge) to the detailed technical implementation in hardware and software.

The automotive domain is known to be different from other domains due to the lack of national and international regulators or certification authorities for functional safety. As such, the standard ISO 26262 does not require a certification by a public authority unlike the aviation industry, where certification is needed. Even if certification is not formally required [18] , car manufacturers use compliance to ISO 26262 as a mean to qualify components and potential suppliers of E/E components (i.e. safety assurance). In this sense, the word certification is often used to describe how a certain system has been developed in compliance with ISO 26262 and audited by an independent assessor. These assessors ensure that the required safety integrity level is achieved by means of:

- Functional safety audits which evaluate the implementation of the processes required for the functional safety activities.
- Compliance management: reviews to check if a selected work product complies with the corresponding ISO 26262 requirements.
- Functional safety assessments that evaluate the functional safety achieved by the item in question.
- GAP analysis: based upon the safety integrity level required for an item, confirm that the needed development activities were performed.

The case study 2 will be based on an advanced driver assistance function (e.g. a traffic jam assistant function allowing highly automated driving of a car on highways up to a defined max speed), in which several electric drives (controller, power electronics and electric machine) act as actuators.

The case study will be executed using modelling, analysis and verification tools and their respective tool integrations.

2.2.2 Technical description of the case study

The components below are the focus of this use-case (this does not mean that all of these components necessarily have to be specified/implemented to the full extend in the context of the case study, as the main purpose of the case study is to demonstrate applicability and benefit of new AMASS technical results):

- ADAS subsystems:
    - Radar (24GHz or 77GHz)
    - Video camera
    - Sensor data fusion
- Vehicle control unit
- Electrical power steering system
- Brake system
- Electric Drive sub-systems:
    - Battery management
    - Main-switch
    - Inverter

As prototype, it is planned to use a demo car provided by B&M. This is currently based on ARM controllers. During AMASS, B&M plans to extend the fleet of demo cars by more models. Since B&M offers engineering services, Infineon is a chip manufacturer in the automotive domain. Thus, it is intended to build one or more B&M demo cars based on Infineon’s AURIX™ microcontroller platform. AURIX is available in different generations, the 1st is available, 2nd coming soon, while 3rd is in an early development phase. So, AMASS will help to collect relevant requirements that can still be considered for future AURIX development.

For the project itself, most likely the AURIX 1st generation will be used. Depending on availability, during the third year the 2nd generation might become a replacement.

KMT will support the case study mainly with tools for modelling the components mentioned above and providing safety analysis for the components and the system. The goal is to demonstrate advanced features of the AMASS project like the seamless integration of different tools that is elaborated in WP5, the architectural pattern approach of WP3 and the re-use approach of WP6. The efficiency advantages gained from such techniques will be demonstrated. The tool based approach will help to support later assessment of the safety case.

In the project, a prototype will be developed. Assessment itself will not take place. However, all necessary steps will be performed to prepare later assessment of products and systems.

2.2.3 Case study state of the art

The focus of development is on building blocks for future ADAS systems. The collaboration within AMASS will support the collection of field data and user requirements.

Currently, many ADAS systems rely on data fusion from two sources, e.g. camera and radar. To handle the amount of raw input data, a pre-processing takes place, usually supported by extra components such as ASICs or FPGAs. One of the goals is to increase integration of growing functionality into fewer components.

Relevant standards need to be considered, such as ISO 26262 for functional safety. Since ISO 26262 second edition most likely will be issued as draft international standard during this year or in 2017, it will be considered as available state of the art.

With the communication between cars and their environments (Car2X), an extra security aspect arises. Today, for security, there is no established standard that compares to ISO 26262 in the automotive domain. Standardization bodies are working on extensions of the standards, but nothing is established and accepted as worldwide de facto standard so far. The only available reference (in automotive) is the “Cybersecurity Guidebook for Cyber-Physical Vehicle Systems” of the SAE J3061. However, security standards exist in other domains (e.g. ED-202A: “Airworthiness Security Process Specification”) and will could serve as additional guideline here.

2.2.4 Case study state of the practice

In AMASS, controllers (e.g. ARM and AURIX) will be used to control the demo cars (provided by B&M). Functionality includes ACC. The traffic jam assistant, for example, will use sensor data (which undergoes data fusion) to control the car with respect to:

- engine control (keep distance to previous car)
- braking system (keep distance to previous car and emergency braking)
- steering (lane keeping support)

The high level part (vehicle level down to specification of the drive component) will be provided by B&M, a company that has experience with e-mobility and driver assistance projects with more than ten German carmakers and Tier1-suppliers, and has defined similar, predecessor case studies in running research projects [16].

The low level part (drive component down to control algorithms and control/power hardware) will be provided by Infineon, who is a leading supplier of both microcontrollers for safety critical vehicle applications (AURIX series) and of gate driver and power semiconductors for electric actuators (e.g. steering systems) and electric traction drives.

The safety case will be provided with the help of tool support, especially for the safety analysis results required by the application of the ISO 26262 standard. Medini analyse will be used for the provision of the HARA (Hazard Analysis and Risk Assessment), the FMEDA (Failure Modes, Effects and Diagnostic Analysis) and the FTAs (Fault Tree Analysis). The system model will be the starting point for the safety analysis and also for the verification/validation.

For security, the situation is more difficult, as no standards or standard tools are available. We can base our work on results of research projects, but there is no real "state of practice".

2.2.4.1 Workflow

For safety, the workflow follows the ISO 26262 safety lifecycle. For security, as there is no real state of the practice, we will follow the proposed workflow of research projects (e.g. SESAMO). An example of a combined safety security workflow looks like in Figure 8.

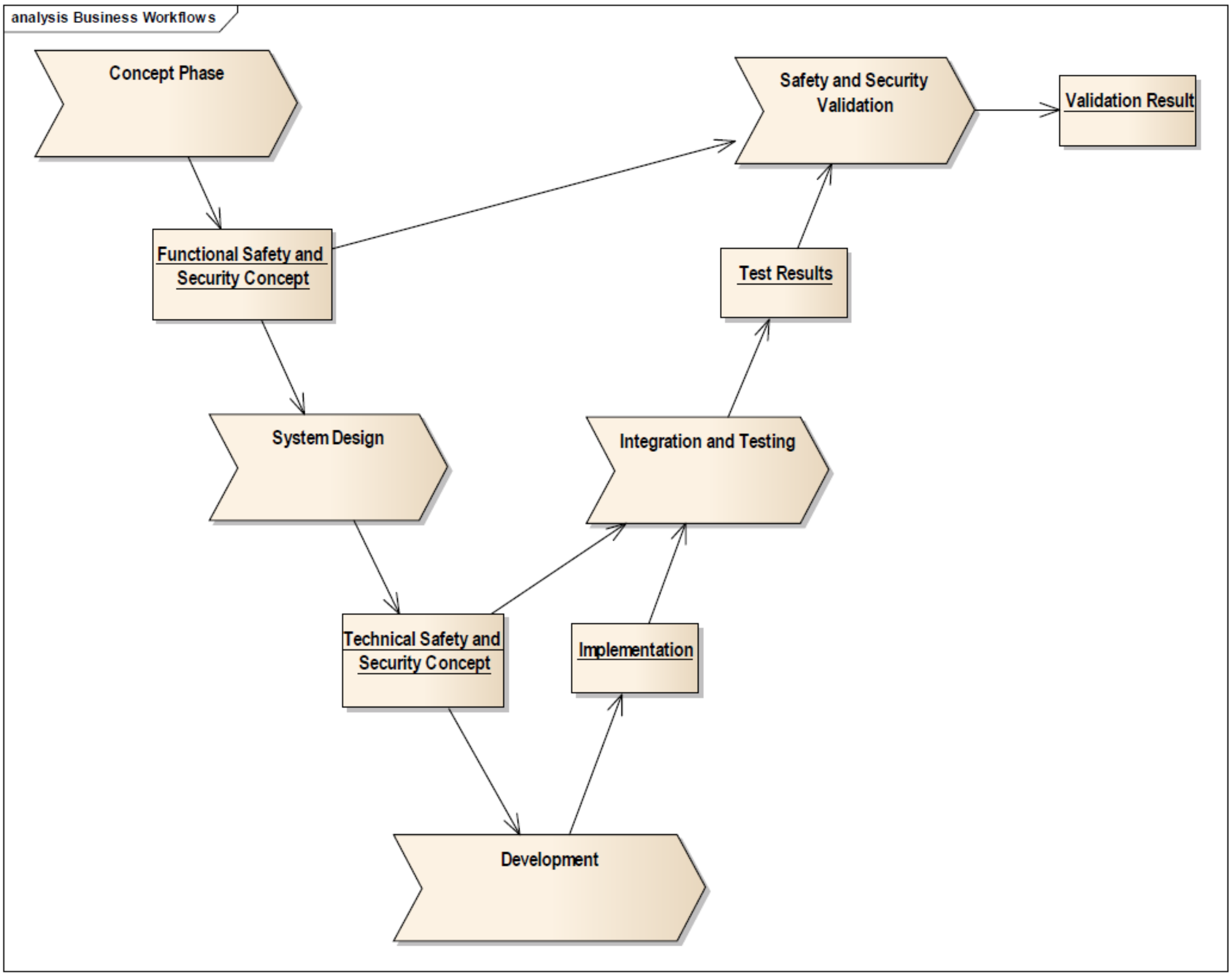

analysis Business Workflows
Concept Phase
Safety and Security Validation
Validation Result
Functional Safety and Security Concept
Test Results
System Design
Integration and Testing
Technical Safety and Security Concept
Implementation
Development

Figure 8. Safety and Security Workflow of the SESAMO project

2.2.4.2 Assessment

The case study serves as demonstrator. The system will not be assessed by a third party. An assessment drill will be organized within the CS-team.

2.2.4.3 Involved roles

The case study demonstrates a complete lifecycle, from OEM, via Tier suppliers to semi-conductors. Most roles involved in such a lifecycle will be part of the case study, although the case study will concentrate on innovative aspects rather than on completeness on each level.

On average, five persons will be involved by IFX in this case study with the following roles:

- Project manager (monitoring the case study development)
- System-architect (engineers in the System Development sub group of the Infineon Automotive Business Group)
- System and safety engineers (Safety Manager of the Automotive Business Group, supported by Quality Management)
- Hardware/Software engineers (HW/SW-development for CS2)

2.2.4.4 Tools and Tool chains

2.2.4.4.1 Used tools and methods (included guidelines)

Tools:

- MATLAB / Simulink for system modeling
- MATLAB Embedded Coder
- Medini analyze (KMT) for safety analysis
- TESTONA

Methods:

- Model-based safety and systems engineering for a distributed networked system
- Contract-based modelling and formulation of requirements

2.2.5 Expected technical improvements

The focus of the CS2 is on the intra domain reuse of safety assurance data. This includes the reuse of safety artefacts such as safety arguments or FTA/FMEAs. The question then arises: what is the safety assurance impact due to design changes? For example, if a component such as a microcontroller is replaced with a model from new product family, re-assessment may become necessary. This additional and often extensive re-assessment effort is likely to be reduced by applying model-based approaches to facilitate reuse during safety assessments, as promoted by AMASS.

This CS will help assess the following objectives: STO1 (System Architecture-driven Assurance), STO3 (Seamless Interoperability) and STO4 (Cross/intra Domain Reuse).

The following improvements within the defined objectives are also purposed.

2.2.5.1 STO1. Architecture-driven Assurance

The use of a model-based approach based on contracts and defined patterns will allow a systematic analysis and creation of functional and technical safety for cooperative system-of-systems during runtime.

2.2.5.2 STO2. Multi-concern Assurance

Safety-Security-Co-Assurance is an important topic. However, this is not the main focus of CS2.

2.2.5.3 STO3. Seamless Interoperability

The case study will show for the involved tools advances in the interoperability. Furthermore, since the scenario covers the whole supply chain, the information exchange among the different stakeholders and inside a team is a topic. If there are tools available in AMASS (e.g. cloud solutions) productivity gains would be demonstrated.

2.2.5.4 STO4. Cross/Intra-Domain Reuse

The case study will demonstrate how library concepts for re-useable components enriched with safety/security information lead to a significant reduction in the effort needed to provide the data for the assurance case.

2.2.6 Business needs

CS2 results will help to reduce assessment efforts in the following cases:

- Change of system components after initial assessment.
- Change of supplier of components involved in the system.

2.2.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

N/A

2.2.6.2 AMASS Goal 2

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*

Primary focus has been put on the automotive domain, however, the systematic approach enables potential users with cross domain challenges to benefit from this intra domain approach as well.

2.2.6.3 AMASS Goal 3

> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

This goal could be addressed by means of the proposed systematic creation of functional and technical safety concepts based on contracts for cooperative system-of-systems and during runtime.

2.2.6.4 AMASS Goal 4

> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

This goal will be addressed by the information exchange among the different stakeholders using the different AMASS solutions.

2.2.7 Usage scenarios

Table 7 and Table 8 show the 7 usage scenarios related to Case Study 2.

Table 7. CS2 IFX usage scenarios

ID:	IFX UsageScenario 1	IFX UsageScenario 2	IFX UsageScenario 3
Related CaseStudy	CS2 and CS3 will establish exchange,	CS2 and CS3 will establish exchange,	CS7
Addressed Domains	automotive	automotive	automotive and aviation
Scenario Name	Re-assessment after component change	Assessment of product families	Cross domain learning between automotive and aviation domain
Short Description	This scenario looks an already assessed automotive system. Later in lifetime, components will be changed (e.g. commercial reasons). So re-assessment is necessary. Goal is to keep the efforts as low as possible.	Product families derive instances which are often 80% identical. How do we need to describe and assess them to ensure minimized efforts for later overall system assessment ?	Aviation has to deliver fail operational systems, automotive so far "only" fail safe. With the trend to autonomous driving, automotive has to learn how to build fail operational systems
Stakeholders	system engineers (keep functionality) as well as both safety (ISO 26262) and security engineers	system engineers (keep functionality) as well as both safety (ISO 26262) and security engineers	system engineers (keep functionality) as well as both safety (ISO 26262) and security engineers
Stakeholder constraints	within the project: none after project end: depends on	within the project: none after project end: depends on	main problem: no multi domain standard for safety and security
Addressed Business Goals:	main focus on G1 and G2	main focus on G2 and G3	main focus on G2 and G3
Process Steps	Product development on system level concerning safety/security -System requirements -Design for Safety - FTA, FMEA -System modelling -System verification	Product development on system level concerning safety/security -System requirements -Design for Safety - FTA, FMEA -System modelling -System verification	Product development on system level concerning safety/security -System requirements -Design for Safety - FTA, FMEA -System modelling -System verification
Concerns	Safety and Security	Safety and Security	Safety and Security
Cross-system certification	N/A	N/A	N/A
Cross-domain certification	no	no	yes
Engineering Environment (Interoperability)	partner specific design flow	partner specific design flow	partner specific design flow
Challenges	N/A	N/A	N/A
Standards	ISO 26262, internal process	ISO 26262, internal process	ISO 26262, DO 178, DO 254
Any wishes for usage scenario	N/A	N/A	N/A
Any known constraints for usage scenario	different design flows are used, not all interfaces are available	different design flows are used, not all interfaces are available	different design flows are used, not all interfaces are available

Remark to US3:

The shift from fail-silent to fail-operational systems poses a great challenge for future automotive systems, since fail-operational behaviour is up to now only implemented in other embedded systems domains with different constraints, such as the cost per unit in avionic systems [17]. However, avionics solutions such as 2oo3 (2 out of 3) are cost prohibitive in automotive. On the one hand, current research is heading into the direction of creating functional safety concepts that provide fail-operational behaviour at system level for individual functions. On the other hand, current microcontroller architectures are also dealing with the challenge of providing fail-operational operations without compromising costs by means of, for instance, intelligent distributed fault supervisors [18].

In the work performed, we had a focus on reuse of COTS components between application domains. Although there are standards addressing the same abstraction level (in this case component) and hold similarities (e.g. “ISO 26262-Part 5: Product development at the hardware level” versus “DO-254RTCA/DO-254, Design Assurance Guidance for Airborne Electronic Hardware”) , the dimensions of requirements are different and prevent an easy 1:1 comparison. Further efforts are necessary to first prioritize and then “translate” them between domains.

In case later work would be extended to system level, extra standards such as ARP4754, ARP4761 must be considered.

Table 8. CS2 TEC and KMT usage scenarios

ID:	TEC Usage Scenario 4	TEC Usage Scenario 5	TEC Usage Scenario 6	KMT UsageScenario 2
Related CaseStudy	CS2	CS2	CS2	CS2
Addressed Domains	Automotive	Automotive	Automotive	Automotive
Scenario Name	Assurance Management Tool	Architectural patterns for assurance	Fault Injection for Safety and Controllability Evaluation of Cooperative System of Systems	
Short Description	1) Compliance with Standards/ product and process assurance management tool to support the compliance assessment 2)Reuse	Creation of arguments for fault tolerance and specific technologies	1) calculate the FTTI which is directly related to the controllability of vehicles 2) evaluate and improve the robustness of automated functions 3)early V&V of safety concepts 4) obtain trade-off evaluation results between safety and cost issues, already at concept level.	Modeling and analysis of the system incl. architecture model, functions, requirements & safety goals + required safety analysis with focus on re-use and exchange of components (e.g. change of HW) (via e.g. component libraries) which can support SEooC
Stakeholders	Safety Manager Assurance Manager Quality Manager Safety Assessor	Safety Engineer System Engineer	System engineer Safety engineer Verification/Test Engineer	Safety Manager Requirements Engineer System engineer Safety engineer
Stakeholder constraints	None	None	None	None
Addressed Business Goals:	G4,G2	G1, G2, G3, G4	G4, G1, G3	G1, G4
Process Steps	N/A	N/A	N/A	Product development on system/HW level concerning safety (security) -System/HW requirements -System/HW design -System/HW analysis -System/HW modelling
Concerns	Safety	Safety	Safety	Safety
Cross-system certification	No	No	No	potentially yes
Cross-domain certification	No	No	No	none
Engineering Environment (Interoperability)	Open Source tools AMASS tools when available for use and evaluation	Open Source tools AMASS tools when available for use and evaluation	Open Source tools AMASS tools when available for use and evaluatio	medini analyze, Rhapsody or EA, any requirement management tool, Office tools
Challenges	Assurance challenges on SoS, architectural/technological patterns	Creation of argument patterns per system component for fault tolerance and specific technologies	Safety and Security co-engineering, FI in SoS, FI addressing ADAS for cooperative systems	re-use of safety analysis results
Standards	ISO 26262, J3061	ISO 26262, J3061	ISO 26262, J3061	ISO 26262
Any wishes for usage scenario	N/A	N/A	N/A	System design done with SysML. Case study owner has to provide HW information and proper design models to support safety analysis
Any known constraints for usage scenario	N/A	N/A	N/A	none

2.3 CS3: Collaborative automated fleet of vehicles

2.3.1 Short description of the case study

The automotive case study "Collaborative automated fleet of vehicles" describes how driving will be in the future. The focus is on the automation of cooperative vehicle functions that allow for the control of selected aspects of longitudinal and lateral motion of a car without driver intervention.

These functions are based on a fused environment model that integrates data from various sensor systems, as well as additional information based on Car2Car and Car2Infrastructure communication. An example of a highly automated networked vehicle function is the Cooperative Adaptive Cruise Control (CACC). CACC enables to connect the vehicles with each other, to negotiate and agree on synchronized value of the speed and warn each other about break interventions or other scenarios. Limits can be further identified and reacted.

In addition, spontaneous networking between vehicles (Car2Car) or between vehicles and infrastructure (Car2Infrastructure) results in modern networked cars, which are more reliable to deal with troublesome situations on the road. The results of the research will increase the safety, give a better traffic flow, and improve the energy efficiency. On the other side new challenges arise, e.g., more open and flexible functions, more difficult safety and security design and the necessity of open interfaces. Furthermore, there is no single manufacturer or system architect who would be fully responsible for the fulfillment of the requirements, in particular, safety requirements. This case study is an ideal application of the AMASS objectives. A demonstrator of the fleet of vehicles constitutes an important part of the validation of the AMASS research results.

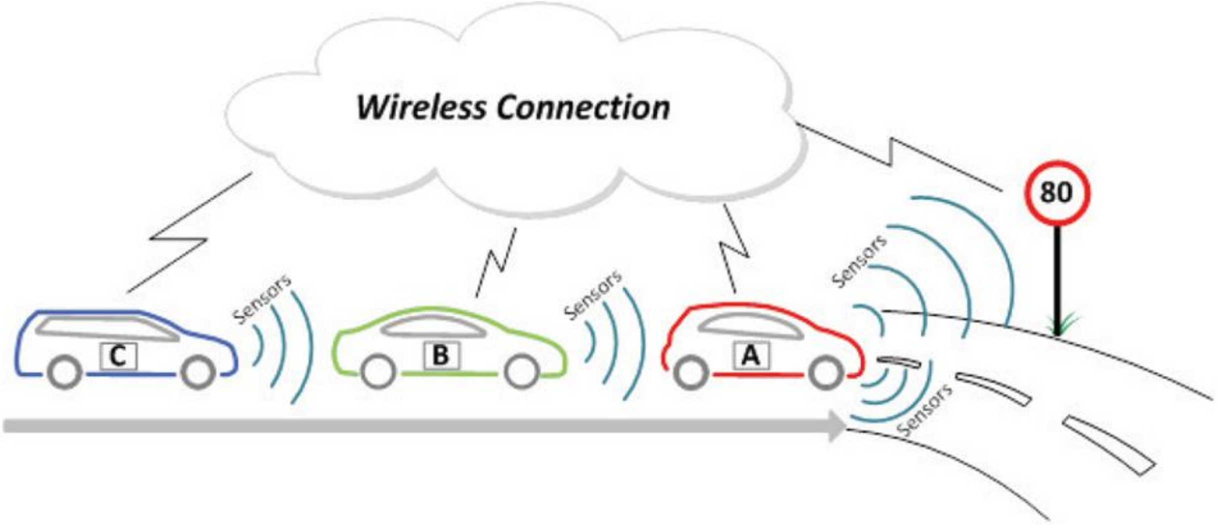


Figure 9. Cooperative Adaptive Cruise Control (CACC)

2.3.2 Technical description of the case study

A collaborative automated fleet consists of multiple vehicles containing actuators and sensors. The actuators are controlled by embedded systems such as ECUs. The controllers are connected with each other in order to establish cooperation. Nowadays, the traffic density is so high that the need of intelligent systems is increased. Traffic flow can be improved by driving at smaller inter-vehicle time gaps using such intelligent systems. An important requirement for the implementation of CACC systems is to have string stable behaviour of a vehicle platoon. The challenge for the string stability is to avoid the amplification of the so-called spacing error (distance error). The spacing error is a result of a delay in the communication. The communication delay and the spacing error accumulate, such that the system becomes instable.

Another important aspect of a string stable platoon is the chosen topology of the platoon. There are different string stability approaches for different platoon topologies. The most common method for vehicle platooning is shown in Figure 10.

a) Designated platoon leader
b) N-Vehicles look ahead

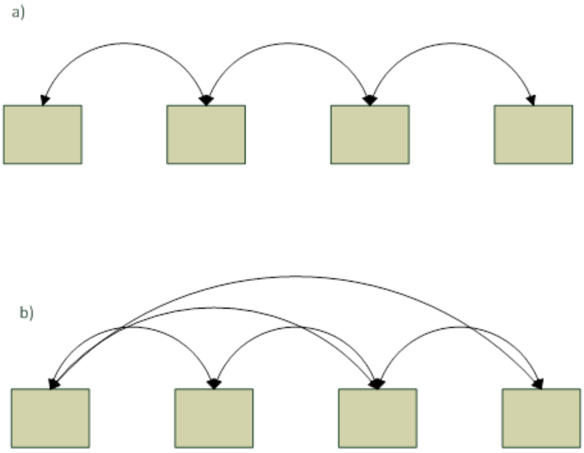


Figure 10. Platoon topologies

Both topologies are bidirectional. This means that the information exchange flows from the preceding vehicle to the end of the platoon and in the opposite direction. Therefore, the vehicles use the inter-vehicle distance measured by the in-vehicle sensors and the acceleration, position and velocity sent by wireless LAN, in case a) from the preceding vehicle, in case b) from the leader and preceding vehicles. This information is then compared from the trajectory modeler and the controller to produce the best controlling results.

For a string stable platoon, it is necessary to have a reliable system.

The communication between the vehicles in a platoon is realized by the standard IEEE 802.11 and IEEE 1609. The wireless stack has a lower layer, a network service and upper layers to communicate with the vehicles. The standard for the communication is the IEEE 802.11 p that was especially designed for vehicular traffic. Wireless communication needs a short medium range technology to communicate with the vehicles. Therefore, the DSRC (Dedicated Short-Range Communication) system is utilized, and it operates in the 5.9 GHz band width.

Data rates of 27 Mbit/s are possible, but the real effective data rate is probably 6 Mbit/s. For the wireless antennas no specifications are given, but the standard organizations should create a common standard. It has to be realized a radius to the side of 300 meter and 1000 meter forward.

There are few important messages in the Car2Car communication field. One of this is the Cooperative Awareness Message (CAM) that allows sharing information with each other without any persistent communication link between the vehicles. The information between vehicles is shared by broadcasting or geocasting to all other surrounding vehicles. On the basis of the shared data between the vehicles, it is possible to analyse the vehicle trajectory and restore traffic patterns. CAM information includes Message ID, position, acceleration, speed, heading and timestamp.

2.3.2.1 Car2X use case scenarios

Vehicle2Vehicle communication is the information exchange between at least 2 vehicles with the aim to determine scenarios that could potentially lead to hazards as well reacting to them earlier than the driver's reaction. Some typical use case scenarios are:

- **Cooperative Forward Collision Warning:** sudden braking is mostly the reason for rear-end collision. The use case of Cooperative Forward Collision Warning is to avoid rear-end collision with other vehicles. Information is constantly being exchanged between the vehicles. This information consists of data such as the position, acceleration speed and heading. The ego vehicle predicts the driving behaviour of all other nearby vehicles based on the listed information. When the ego vehicle detects a critical action, the ego vehicle reacts and warns the driver.
- **Hazardous Location V2V Notification:** uses the shared information of preceding vehicles to determine hazardous locations. The hazardous road characteristics like sharp bends, ice, aquaplaning or other obstacles can be detected by the vehicle sensors (e.g. sensors used for Electronic Stability Program). This information can be shared with any number of vehicles over a wide area. Also, information can be shared from external service providers via roadside units.
- **Pre-Crash Sensing/warning:** uses periodically shared information from adjacent vehicles to predict a collision.

Figure 11 depicts an example of a state machine for graduation and degradation of a CACC system. In the CACC state, the vehicle performs the cooperative drive automatically. After a successful connection, the vehicles keep the CACC headway time. When the state manager does not receive a signal, it degrades the CACC to ACC such that the distance is kept constant by using the conventional distance sensor. Obviously, this leads to larger time gaps than the ones achieved by the CACC.

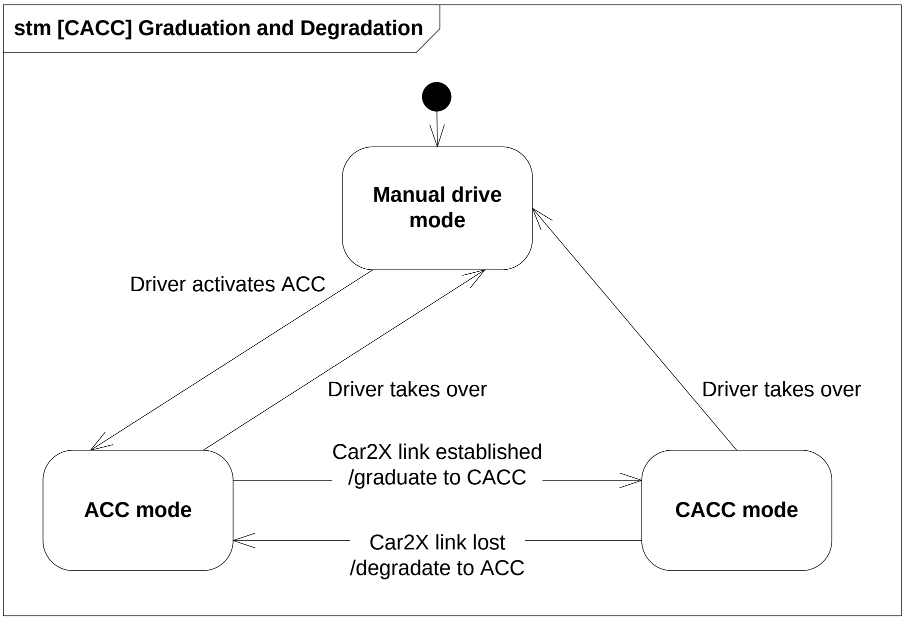

stm [CACC] Graduation and Degradation
Manual drive mode
Driver activates ACC
Driver takes over
Driver takes over
ACC mode
CACC mode
Car2X link established /graduate to CACC
Car2X link lost /degradate to ACC

Figure 11. State Machine Diagram for graduation and degradation of assistant systems

Figure 12 illustrates the functional architecture of a CACC system. The W-LAN connection is responsible for the communication data management. The CACC-controller controls vehicle motion based on the inter vehicle distance of the platoon members.

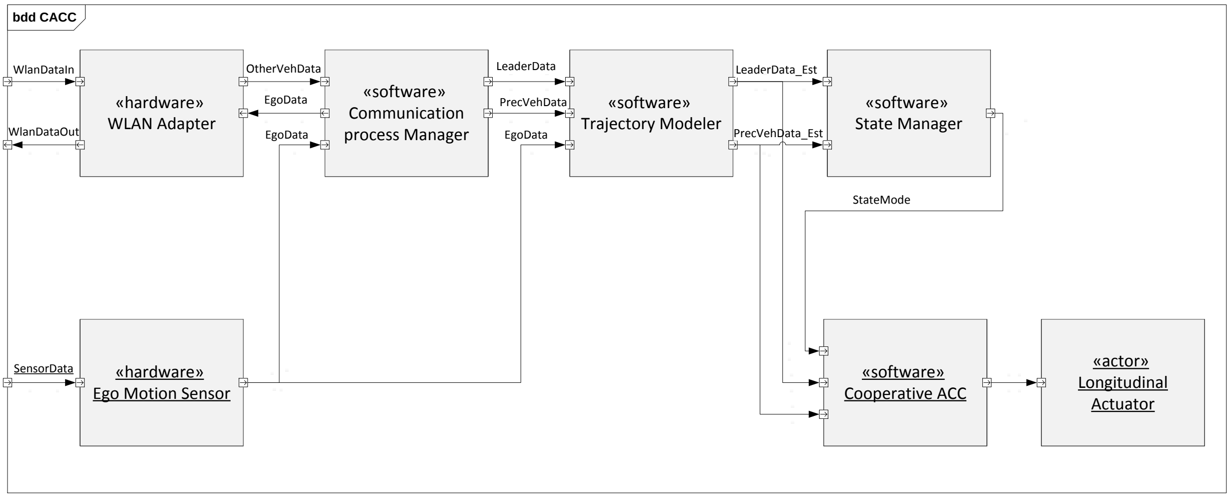


Figure 12. Functional architecture of CACC

2.3.3 Case study state of the art

The Car2Car Communication Consortium Manifesto [1] is the detailed definition from the Consortium. It defines all coming standards for vehicles to communicate with each other.

A reference for the string stability is the research "Design and Experimental Evaluation of Cooperative Adaptive Cruise Control" [2]. It shows the control design and error dynamics by Cooperative systems. They tested a real fleet for the stability and the wireless communication.

Other relevant contributions are, e.g.:

- Analysis and design of controllers for cooperate and automated driving (Jeroen Ploeg) [3].
- Vehicular-2-X Communication (Radu Popescu-Zeletin, Ilja Radusch, Mihai Adrian Regani) [4].

2.3.4 Case study state of the practice

Vehicles today turn around with standards like antilock braking system, electronic stability program, attention assist and many more. The book “Handbuch Fahrerassistenzsysteme” [5] describes many ADAS systems that are actually included in vehicles today.

Another innovation in the Car2X field is presented by the new Mercedes Benz E-Class. The new E-Class is able to communicate with building sites on the road and to share information with other new E-Classes. The exchanged information includes the position of the building sites [6].

Leading projects are conducted by the pioneers of the platooning field, Mercedes Benz. In one project, three trucks were tested successfully by Mercedes Benz on a German highway A61 with a – by Daimler developed – connected Highway Pilot [7]. The video in [8] shows the implemented track platooning and the related use cases in practice. The company Continental belongs also to the leading companies for truck platooning, with many successfully performed tests.

Highly or fully automated vehicles, which are today in a prototype stage (but partly involving testing on public roads), heavily rely on infrastructure data, which is today restricted mainly to GPS and partly traffic information broadcasted by available radio systems, to some extend also communication to proprietary back-end servers. It can be expected, that these highly or fully automated vehicles would benefit from car-to-car communication as well (if the car-to-car communication is not already part of the automation function itself, as is the case in platooning or cooperative ACC settings) – but this is hampered today by the lack of a sufficient amount of Car2X enabled vehicles on the roads today.

As there are no series vehicles on the market equipped with highly automated and networked functions, it is not possible to determine the state of the practice regarding design, safety analysis and safety case generation for such systems. The AMASS project, in particular as applied to Case Study 3, is part of a large stream of industrial and academic research projects aiming at defining the state of practice for the future.

Nevertheless, already today it is possible to derive requirements for the future solution and to extrapolate the current state of practice in development of highly automated and/or networked vehicle function. As the author of this section has worked as a consultant with 10+ major carmakers and automotive suppliers, there is a sound and consistent impression of vehicle function development in industrial practice: formally, the vast majority of carmakers and suppliers adheres to the V-model as defined e.g. in the Automotive SPICE standard ISO 15504 and in ISO 26262 for safety-related systems. But even for traditional mechatronic systems, the distribution of the V-model process across carmaker and one or several suppliers is not sufficiently mastered, in particular when it comes to the integration of safety analyses and the safety case across different companies, or when it comes to reusing pre-existing components in new contexts. Proposals that have been elaborated in cooperation with industry partners do exist, but have not yet made their way into practice in a broad range [21][22]. Using contracts as a mean of doing so has been proposed by the author to different industry companies, and attracted some attention, but not yet practices in any project to the author’s knowledge.

Regarding runtime-safety certification, which would be required for cooperative systems (systems-of systems, that constitute themselves at runtime) there is no existing approach throughout the industry, to the author’s best knowledge. As many of these innovative functions make their way directly from corporate research departments into test vehicles, even the V-model and ISO 26262 have sometimes not been observed so far and it seems a challenge to compensate for this when taking the new functions into series vehicles requiring safety-certification. The approach of using runtime contracts for safety certification of systems-of-systems in the automotive domain, as discussed in the AMASS consortium, has never been applied in industry, nor is there any other commonly accepted approach to address the same tasks.

2.3.4.1 Workflow

Table 9. CS3 Workflow

DEVELOPMENT PHASE	ACTIVITY
Case Study Specification	Specification of the necessary artefacts
	Requirement specification
	Item definition & Architecture design
	Hazard analysis and risk assessment
Case Study Implementation and Benchmarking	Implementation of a physical platform of 3 model cars
	Benchmarking for a CACC enabling to group the vehicles into platoons at runtime

During each phase the following process steps are taken into consideration:

- System/SW/HW requirements (based on contracts)
- System/SW/HW design
- System/SW/HW analysis (based on contracts)
- System modelling
- System/SW/HW verification

2.3.4.2 Assessment

Assessments will be done by persons within the participating organizations with the corresponding competence.

2.3.4.3 Involved roles

6-7 persons will be involved by B&M in this case study with the following roles:

- Project manager (monitoring the case study development)
- System-architect (designing the architecture of the system)
- System and safety engineers (design-concept for the case study and model cars including safety aspects)
- Hardware/Software engineers (HW/SW-development for the CS)

2.3.4.4 Tools and Tool chains

2.3.4.4.1 Used tools and methods (included guidelines)

Tools:

- MATLAB / Simulink
- MATLAB Embedded Coder
- Microsoft Visual Studio
- Savona (to be developed tool within AMASS project)

Methods:

- Model-based safety and systems engineering for a distributed networked system
- Contract-based modelling and formulation of requirements
- Safety assurance during runtime based on the cooperation of the members in the network

2.3.4.4.2 Tool chain

For safety analysis and requirements management currently, only Visio (ViConEx) and Excel are used. AMASS tools and methods will be applied in the course of the project; especially the tool Savona, which is under development right now. This will be used for modelling and architecture design and formulation of semi-formal contracts. Furthermore, MATLAB/Simulink is used for modelling of the system and controller design purposes. MATLAB Embedded Coder is used for automatic code generation.

2.3.5 Expected technical improvements

The focus of the CS3 is on the automation of cooperative vehicle features, which enables a partly autonomous longitudinal and lateral control of vehicles in a network and during runtime. This CS will help assess the following objectives: STO1 (System Architecture-driven Assurance), STO2 (Multi-concern assurance), and STO3 (Seamless Tool Interoperability).

The following improvements within the defined objectives are also purposed:

2.3.5.1 STO1. Architecture-driven Assurance

The use of a model-based approach based on contracts and defined patterns will allow a systematic analysis and creation of functional and technical safety for cooperative system-of-systems during runtime.

2.3.5.2 STO2. Multi-concern Assurance

Formulation of different types of concerns via contracts is possible and the verification and validation of requirements by checking the contracts will be improved.

2.3.5.3 STO3. Seamless Interoperability

A platform for the evaluation of the objectives based on the running car instances and the interactions of multiple autonomous cars interplaying with each other will be provided.

2.3.5.4 STO4. Cross/Intra-Domain Reuse

N/A

2.3.6 Business needs

CS3 along with the results from other WPs could cover all the AMASS goals with reservations regarding the mentioned metrics in the goals. This is to still to be analysed and found out within the evolution of the project.

2.3.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

This goal is addressed within CS3 via the proposed model-based safety and systems engineering methods based on contracts for a distributed networked system (system-of-systems) and via the development of advanced driver assistance systems (e.g., Traffic Sign Recognition (TSR) systems).

2.3.6.2 AMASS Goal 2 and Goal 3

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*
>
> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

These goals could be addressed by means of the proposed systematic creation of functional and technical safety concepts based on contracts for cooperative system-of-systems and during runtime.

2.3.6.3 AMASS Goal 4

> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

This goal can be addressed by the introduced development of advanced driver assistance systems for cooperative systems (e.g., Cooperative Adaptive Cruise Control) and their physical implementation and also by the validation and verification of the developed methods and tools (e.g., Savona) within the AMASS project.

2.3.7 Usage scenarios

Table 10, Table 11, Table 12 and Table 13 show the 16 usage scenarios related to Case Study 3.

Table 10. CS3 B&M usage scenarios

ID:	B&M UsageScenario 1	B&M UsageScenario 2	B&M UsageScenario 3	B&M UsageScenario 4
Related CaseStudy	CS3	CS3	CS3	CS3
Addressed Domains	Automotive	Automotive	Automotive	Automotive
Scenario Name	CS3Modelling	CS3Safety	CS3SafetyRuntime	CS3ADAS
Short Description	Model-based safety and systems engineering based on contracts for a distributed networked system (system-of-systems)	Systematic creation of functional and technial safety concepts based on contracts	Systematic creation of functional and technial safety concepts based on contracts for cooperative system-of-systems during runtime	Development of advanced driver assistance systems (e.g., Traffic Sign Recognition (TSR) systems)
Stakeholders	System engineers Safety engineers	Safety engineers	Safety engineers	System engineers
Stakeholder constraints	None	None	None	None
Addressed Business Goals:	G1 (with reservations reg. The metrics)	G2, G3 (with reservations reg. The metrics)	G2, G3 (with reservations reg. The metrics)	G1, G3 (with reservations reg. The metrics)
Process Steps	Product development on system level concerning -System requirements (based on contracts) -System design -System analysis (based on contracts) -System modelling -System verification	Product development on system level concerning -System requirements (based on contracts) -System design -System analysis (based on contracts) -System modelling -System verification	Product development on system level concerning -System requirements (based on contracts) -System design -System analysis (based on contracts) -System modelling -System verification	Product development on system level concerning -System requirements (based on contracts) -System design -System analysis (based on contracts) -System modelling -System verification
Concerns	Safety Systems Engineering	Safety Systems Engineering	Safety Systems Engineering	Safety Systems Engineering
Cross-system certification				
Cross-domain certification				
Engineering Environment (Interoperability)	MATLAB / Simulink Savona (to be developed tool within AMASS project)	Savona (to be developed tool within AMASS project)	Savona (to be developed tool within AMASS project)	MATLAB / Simulink Savona (to be developed tool within AMASS project)
Challenges	Contract-based modelling and formulation of requirements, Safety assurance during runtime and based on the cooperation of the members in the network	Contract-based modelling and formulation of requirements, Safety assurance during runtime and based on the cooperation of the members in the network	Contract-based modelling and formulation of requirements, Safety assurance during runtime and based on the cooperation of the members in the network	Contract-based modelling and formulation of requirements, Safety assurance during runtime and based on the cooperation of the members in the network
Standards	ISO 26262	ISO 26262	ISO 26262	ISO 26262
Any wishes for usage scenario	N/A	N/A	N/A	N/A
Any known constraints for usage scenario	N/A	N/A	N/A	N/A

Table 11. CS3 B&M and TEC usage scenarios

ID:	B&M UsageScenario 5	B&M UsageScenario 6	TEC Usage Scenario 7	TEC Usage Scenario 8	TEC Usage Scenario 9
Related CaseStudy	CS3	CS3	CS3	CS3	CS3
Addressed Domains	Automotive	Automotive	Automotive	Automotive	Automotive
Scenario Name	CS3CooperativeSystems	CS3V&V	Assurance/Certification Management Tool	Fault Injection for Safety and Controllability Evaluation of Cooperative System of Systems	Architectural patterns for assurance
Short Description	Development of advanced driver assistance systems for cooperative systems(e.g., Cooperative Adaptive Cruise Control) and their physical implementation	Validation and verificationof the developed methods and tools (e.g., Savona) within AMASS-project	Compliance with Standards/ product and process assurance/certification management tool to support the compliance assessment and certification	1) calculate the FTTI which is directly related to the controllability of vehicles 2) evaluate and improve the robustness of automated functions 3)early V&V of safety concepts 4) obtain trade-off evaluation results between safety and cost issues, already at concept level.	Creation of arguments for fault tolerance and specific technologies
Stakeholders	System engineers Software engineers	System engineers Safety engineers	Safety/Security Manager Assurance Manager Quality Manager Safety Assessor	System engineer Safety engineer Security engineer Verification/Test Engineer	Safety/Security Engineer System Engineer
Stakeholder constraints	None	None	None	None	None
Addressed Business Goals:	G4 (with reservations reg. The metrics)	G4 (with reservations reg. The metrics)	G4	G4, G1, G3	G1, G2, G3, G4
Process Steps	Product development on system level concerning -System requirements (based on contracts) -System design -System analysis (based on contracts) -System modelling -System verification	Product development on system level concerning -System requirements (based on contracts) -System design -System analysis (based on contracts) -System modelling -System verification	N/A	N/A	N/A
Concerns	Safety Systems Engineering	Safety Systems Engineering	Safety and Security	Safety and Security	Safety and Security
Cross-system certification	electric vehicle sub-system (CS2)		No	No	No
Cross-domain certification			No	No	No
Engineering Environment (Interoperability)	MATLAB / Simulink Savona (to be developed tool within AMASS project)	Savona (to be developed tool within AMASS project)	Open Source tools AMASS tools when available for use and evaluation	Open Source tools AMASS tools when available for use and evaluatio	Open Source tools AMASS tools when available for use and evaluation
Challenges	Contract-based modelling and formulation of requirements, Safety assurance during runtime and based on the cooperation of the members in the network	Contract-based modelling and formulation of requirements, Safety assurance during runtime and based on the cooperation of the members in the network	Certification challenges on SoS, architectural/technological patterns	Safety and Security co-engineering, FI in SoS, FI addressing ADAS for cooperative systems	Creation of argument patterns per system component for fault tolerance and specific technologies
Standards	ISO 26262	ISO 26262	ISO 26262, J3061	ISO 26262, J3061	ISO 26262, J3061
Any wishes for usage scenario	N/A	N/A	N/A	N/A	N/A
Any known constraints for usage scenario	The implentation is planned to be an indoor fleet of 3 small-sized vehicles.	N/A	N/A	N/A	N/A

Table 12. CS3 KMT, MDH, AIT and ViF usage scenarios

ID:	KMT UsageScenario 3	MDH UsageScenario 1	AIT UsageScenario 2	ViF UsageScenario 1
Related CaseStudy	CS3	CS3	CS3	CS3
Addressed Domains	Automotive	Automotive	Automotive	Automotive
Scenario Name		Model-Based Multi-concern Analysis and Assurance Case Generation	Safety & security co-analysis of autonomous and collaborative vehicles	MBS3E
Short Description	Modeling and analysis of the system incl. architecture model, functions, requirements & safety goals + required safety analysis with focus on re-use (e.g. component libraries) and multi concern aspects (especially security). The CS3 is based on CS2.	Support for Safety and Security Analysis Argumentation (SACM/GSN) Contract-based assurance cases Configuration-aware contracts Product-based reuse	Safety and Security co-analysis, co-design following - Safety (ISO26262) - Security (J3061)	Model-based System, Safety, and Security Engineering Support for Safety and Security Analysis (FMEA & FMEVA and FTA & ATA) Contract-based Design for System Architecture by Safety and Security Contracts
Stakeholders	Safety Manager Requirments Engineer System engineer Safety engineer Security engineers	N/A	Risk analyzer Safety engineer Security engineer	G1, G4
Stakeholder constraints	None	N/A	None	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification -System argumentation
Addressed Business Goals:	G1, G4	G1, G2	G1, G2, G3, G4	System engineer Safety engineer Security engineer
Process Steps	Product development on system/HW level concerning safety and security -System/HW requirements -System/HW design -System/HW analysis -System/HW modelling	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System argumentation	Development process, mainly focusing on requirement elicitation, support of implementation of combined safety&security concept	None
Concerns	Safety & Security	Safety and Security	Safety and Security	Safety and Security
Cross-system certification	potentially yes	Yes	No	•Control Vehicle by Smartphone Application(e.g. Intelligent battery charging) •ECU updating over-the-air(e.g. Update of battery management unit)
Cross-domain certification	none	No	No	No
Engineering Environment (Interoperability)	medini analyze, Rhapsody or EA, any requirement management tool, Testing tools, Office tools	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools-Safety/Security Analyses Tool and V&V Tools + industry-required tools when appropriate	safety and security analysis tool; assurance workflow engine (?tbd)	MBSE by EA Tool interaction MBSE Tools-Safety/Security Analyses Tool
Challenges	multi concern design, tool interoperability	Safety and Security co engineering, Commonality & Variability systematization	Safety and security co-engineering and interaction points	Safety and Security co engineering
Standards	ISO 26262	ISO 26262	ISO 26262 SAE J3061	ISO 26262 SAE J 3061
Any wishes for usage scenario	System design done with SysML. Case study owner has to provide models and requirements to support safety analysis	Reuse of established safety methods for security topic	Offer the possiblity to conduct security&safety tests on the intended demonstrator, e.g. inject threats and demonstrate correct, e.g. safe, system reaction	Reuse of established safety methods for security topic
Any known constraints for usage scenario	none	Not so far	Restricted size (max. 3 vehicles) will not allow load tests and may hide issues with scaling.	Only certified tools in the company allowed

Table 13. CS3 ViF and A4T usage scenarios

ID:	ViF UsageScenario 2	ViF UsageScenario 3	A4T UsageScenario 2
Related CaseStudy	CS3	CS3	CS3
Addressed Domains	Automotive	Automotive	Automotive
Scenario Name	S2oPL	P2S2A	Safety & security co-analysis of autonomous and collaborative vehicles
Short Description	Safety- and Security-oriented Process Line Process Modelling Framework that supports - QualityManagment (ASPICE), -Safety (ISO26262) and - Security (J3061) Aspects	Process- and Product-based Safety and Security Assurance • Safety and Security Argumentation Modelling in GSN • Process and Product-based Arguments • Patterns support reuse of argumentation	Model Based Safety Analysis (MBSA) with safety demonstration
Stakeholders	G2	G2, G3	System engineer Safety engineer Security engineer
Stakeholder constraints	Supporting of process management for company and project specific aspects	Safety and security argumentation on concept and system level	None
Addressed Business Goals:	Process manager Safety engineer Security engineer	Quality Assurance Manager Safety engineer Security engineer	G1 - G2 - G3
Process Steps	None	None	Product development on system level concerning safety/security -System requirements -System design -System argumentation
Concerns	Safety and Security	Safety and Security	Safety and Security
Cross-system certification	•Control Vehicle by Smartphone Application(e.g. Intelligent battery charging) •ECU updating over-the-air(e.g. Update of battery management unit)	•Control Vehicle by Smartphone Application(e.g. Intelligent battery charging) •ECU updating over-the-air(e.g. Update of battery management unit)	Between system components
Cross-domain certification	No	No	No
Engineering Environment (Interoperability)	WEFACT(AIT), EPF-C	Prosurance(TEC); PTC-Integrity	MBSE approach + Safety Architect
Challenges	Safety and Security co engineering	Safety and Security co engineering	Safety and Security co engineering - collaborative work between system architect and safety and security analysts
Standards	ISO 26262 SAE J 3061 ASPICE	ISO 26262 SAE J 3061 ASPICE	Iso 26262
Any wishes for usage scenario	Reduce engineering process activities for safety and security engineering	Reusable argumentation patterns for safety and security aspects	N/A
Any known constraints for usage scenario	Different standards use different terms and definitions --> Harmonisation	no joint approach for common safety and security assessment	N/A

2.4 CS4: Design and safety assessment of on-board software applications in Space Systems

2.4.1 Short description of the case study

The space domain case study CS4 aims to evaluate the current processes of On Board Software (OBSW) applications design and safety analysis by comparing them with a model-based design approach adding specific safety characteristics. The Space Agencies and the Space Industrial effort are focusing on advancing the design of critical OBSW applications towards a model-based approach. This approach shall allow the Software and Safety Engineers:

1. To include dependability and safety aspects early in the development process
2. The reuse of components from one mission to another
3. To reduce time and costs, increasing development efficiency

This use case will be based on the OBSW developed for the Ocean & Land Colour Instrument (OLCI) used in the satellite Sentinel-3.

2.4.1.1 Sentinel-3 description

The Sentinel-3 satellite is part of the European Commission’s Copernicus programme, an environmental monitoring programme that tackles the effects of climate change and safeguard everyday lives.

The Sentinel-3 satellite will measure Earth’s oceans, land, ice and atmosphere to monitor and understand large-scale global dynamics. It will also provide information in near-real time for ocean and weather forecasting.

This mission is based on two identical satellites orbiting in constellation. The first satellite (Sentinel-3A) was launched on 16 February 2016 capturing the first images two weeks after the launch [9].

Figure 13. Iberian Peninsula [9]

2.4.1.2 OLCI description

The Ocean & Land Colour Instrument (OLCI) is a multi-spectral optical camera for Ocean and Land Colour. The Instrument Control Module (ICM) is part of the OLCI Electronics Unit (OEU). ICM is mainly responsible for global managing the OLCI elements and supporting the OEU ICM SW, which runs on an ERC32 microprocessor with SPARC v7 architecture.

The OEU ICM SW running is divided in two parts:

- Basic Software (BSW): provides bootstrap and I/O drivers.
- Application Software (APPSW): implements the mission functionality.

2.4.2 Technical description of the case study

This case study will focus on some functionalities of the Application Software (APPSW) of the OEU ICM SW. Namely, the OEU ICM SW implements the algorithm for controlling the Video Acquisition Module (VAM) and the Focal Plane Assembly (FPA), that provide the Science Video Frames for creating the Science Report, which are part of the equipment telemetry.

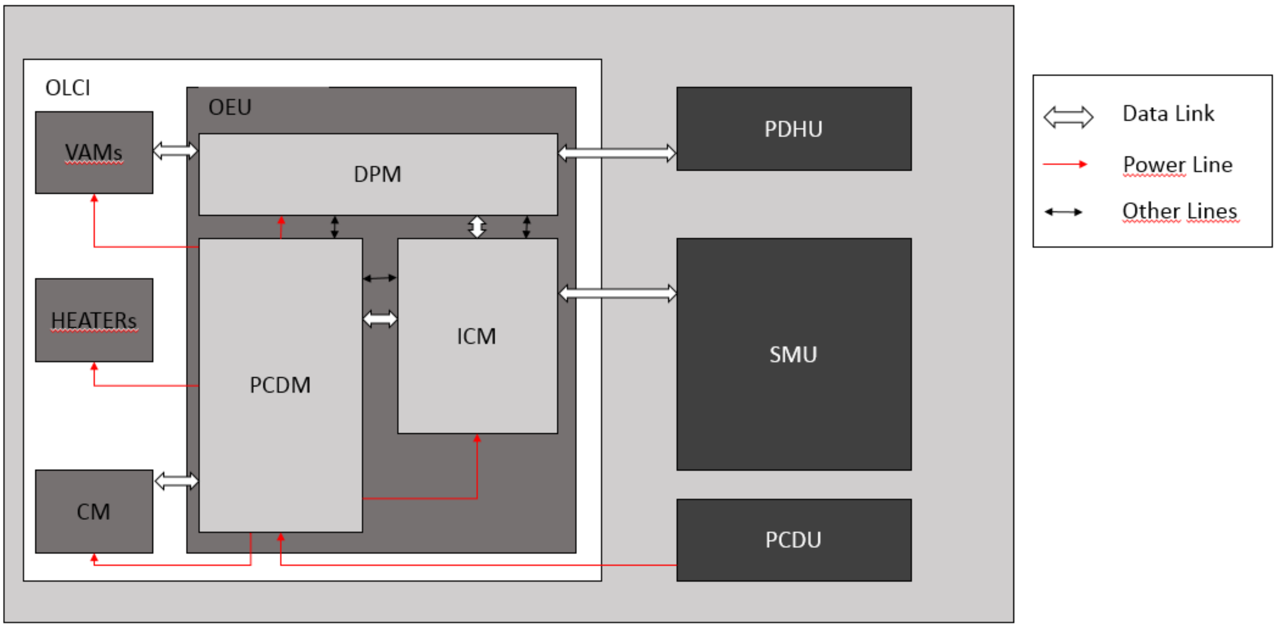


Figure 14. OLCI instrument

2.4.2.1 Technical description of the "Component Reuse"

The CS4 activities will aim to identify software Building Blocks which can be reused in different missions. This shall cover:

- The way the Building Block is defined (e.g., inputs, outputs)
- The non-functional parameters that can be configured (in particular the safety parameters) and allow the SW Engineer to tailor the Building Block according to the requirements of a mission.

The CS4 will identify potential Building Blocks and will assess the feasibility to be used in different operational missions.

2.4.2.2 Technical description of the "Re-qualification"

During the CS4, the impact of a software re-qualification due to a change in the execution platform (e.g., the processor, the communication buses, memory type or size, etc.) will be measured.

Currently the ICM has the following main elements:

- ERC32 processor with PROM, EEPROM and SRAM
- 1553 bus coupler
- RS-422

- On-board Time (OBT) register

2.4.2.3 Technical description of the "Safety analysis with the AMASS platform"

Currently, safety assessment is manually performed. The CS4 will conduct different safety analyses based on the software model. This implies that the safety information has to be present in the model.

The benefits of taking into account the safety information (requirements and constraints) in the early phases of a Software development will be assessed.

2.4.3 Case study state of the art

Standardization is one of the key factors for the ESA (European Space Agency) in order to develop space critical applications. Therefore, the ESA created a series of standards named ECSS (The European Cooperation for Space Standardization) which include standards (normative), handbooks (non-normative) and technical memorandum (non-normative). They include processes for:

- Space project management
- Space product assurance
- Space engineering

The ECSS also addresses dependability and safety processes at system and software level. Here are listed some of these standards:

- ECSS-Q-ST-30 defines the requirements for a dependability assurance programme for space projects.
- ECSS-Q-ST-40 includes the safety programme and the technical safety requirements for critical applications.
- ECSS-E-ST-40 states the principles and requirements applicable to space software engineering.
- ECSS-Q-ST-80 defines the software product assurance programme for space projects. It assesses the need of a SW critical analysis and measures establishment for assuring the reliability of critical software.

The standards are tailored according to the specific needs and requirements of each project and guide the activities to be performed.

2.4.3.1 Model-based design and safety assurance

Nowadays the complexity and functionality of safety critical systems is increasing more and more. This leads to a higher demand of safety and dependability software components for Safety Critical Systems and new software design paradigms.

In this scenario, the ESA is evaluating requirements, techniques and methods to develop space applications based on:

- Model-driven architectures
- Component-based design
- Decoration of models with non-functional attributes
- Modelling both the static and dynamic architecture
- Execution of analysis (e.g., performance, safety, dependability) at model-level
- Code generation from system models
- Qualification

2.4.4 Case study state of the practice

The space use case CS4 focuses on the requirements specification, design and model analysis phases. In the real Sentinel-3 mission these phases were implemented as explained below:

- **Requirements specification**. The requirements were defined and compiled using DOORS and expressed using natural language. Apart from the formulation of the system and software requirements, DOORS included not only the trace among them but also the trace to the design entities and test cases. Furthermore, DOORS stored the FMEA (Failure Modes and Effects Analysis) information and the links among the failure modes/compensation provisions and the corresponding requirements, design entities and test cases.
- **Design**. The design was developed using Borland Together tool and HRT-UML (Hard-Real Time – Unified Modelling Language). This language defines an extension profile of UML. It is used to model generic architectures and it is especially useful for modelling hard-real time systems.
- **Model analysis**: Taking advantage of HRT-UML, the CPU load and schedulability analyses were directly performed. The method selected to analyse the feasibility of the proposed architecture was the Response Time Analysis.

  Despite using a model-driven approach for the architecture design, the design did not include safety information (e.g., safety contracts). The FMEA tables were manually generated and stored in DOORS. Only the FMEA traces to requirements and design entities were automatically generated. The rest of the information provided in the Safety Critical Analysis Report (SCAR) was prepared without any tool support.

Table 14 includes the development phases covered by the space use case CS4 highlighting the tools used in each of them.

Table 14. State of practice of Sentinel-3 OEU ICM SW

DEVELOPMENT PHASE	ACTIVITY	TOOLS USED
Requirements Specification (Requirements baseline and technical specification)	Specification	DOORS
	Traceability	DOORS
Architecture Design / Detail design	Static design	HRT-UML
	Dynamic behaviour	Borland Together (sequence diagrams)
Model Analysis	Schedulability analysis	HRT-UML
	CPU load	HRT-UML
	FMEA/FTA	Manual

2.4.4.1 Workflow

The following figure shows the standard software lifecycle for space operational projects.

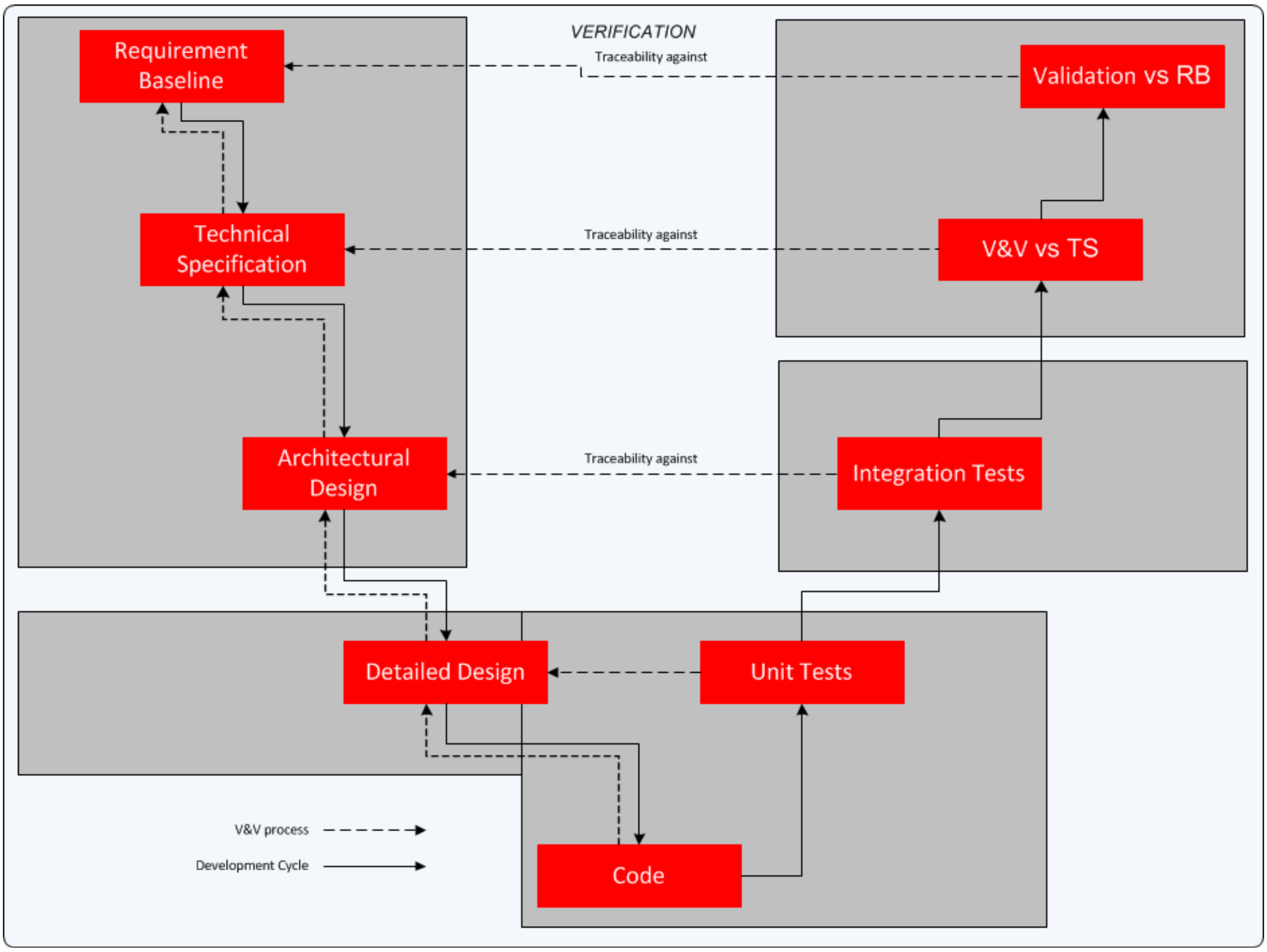


Figure 15. Development life-cycle

The dependability and safety analyses run either in parallel or in conjunction with the software development process. To carry out an efficient software dependability and safety analysis two different techniques can be used:

- Inductive Bottom-Up approach: from identified software failures, the hazards that could arise from them are assessed (e.g. SFMECA technique).
- Deductive Top-Down approach: starts analysing identified hazards-events with the purpose of finding out the potential causes (e.g. SFTA technique).

2.4.4.2 Assessment

These are the different assessment of the embedded operational projects:

- Verification report
- Validation report
- Independent verification and validation process (depending on the criticality level)
- Software critical Analysis report
- Budget report

2.4.4.3 Involved roles

These are the main roles involved in the assessment process:

- System Engineer
- Software Engineer
- Avionics Engineer
- Safety Engineer
- Test or V&V Engineer

2.4.4.4 Used tools and methods (included guidelines)

2.4.4.4.1 Tool chain

Table 14 includes the tools used for the development of the ICM instrument.

Other tools commonly used in Space Systems are the following:

- Requirements specification: DOORS, Requisitepro.
- Design/Development: Eclipse, Enterprise Architect, Rational Software Modeller, MagicDraw, SolidWorks, Qt, Tcl, Oxygen, Matlab, Gcc, Gdb, J2SE, etc.
- Tests: AdaTest, Cppunit, Cxxtesst, VectorCaset, etc.

2.4.5 Expected technical improvements

The following objectives are pursued:

- Assess the feasibility of components reuse (e.g., using different execution platforms).
  Evaluation of how the software components shall be configured (i.e., variants in the product line) to be reused from one mission to another, as well as assess the cost and time reduction during the re-qualification process.
- Analyse, at model level, the impact on the re-qualification when the hardware platform is modified (e.g., partitions, multicore, etc.).
- Analyse the system safety, performance, reliability and availability requirements using the AMASS platform to:
  - Evaluate its applicability in the Space Domain (compliance with ECSS standards).
  - Comparison to current process and practices.
  - Proposals for improvements.

2.4.5.1 STO1. Architecture-driven Assurance

- Introducing the safety concerns in the early phases of the Software development (i.e., software design) will reduce the time for qualifying the final product.
- The use of a model driven approach will facilitate the design and Software reuse and less time will be needed for re-qualification that Software.

2.4.5.2 STO2. Multi-concern Assurance

- The Verification and Validation of safety-related requirements will be improved.

2.4.5.3 STO3. Seamless Interoperability

- Feedback from the activities and tools will be provided to the technical work packages for completing the AMASS platform.

2.4.5.4 STO4. Cross/Intra-Domain Reuse

- N/A

2.4.6 Business needs

2.4.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

Methodology and Model driven design approach to create safety case (ECSS standards based).

2.4.6.2 AMASS Goal 2

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*

Identification of space building blocks which can be tailored and reused from one mission to another.
Generation of safety evidences for space building blocks.

2.4.6.3 AMASS Goal 3

> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

To check different architecture choices at design level (e.g., architectural trade-offs based on the model analysis results).

2.4.6.4 AMASS Goal 4

> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

Integration of all development tools in a single environment, providing fully interoperability among them.

2.4.7 Usage scenarios

Table 15 and Table 16 show the 9 usage scenarios related to Case Study 4.

Table 15. CS4 GMV and TEC usage scenarios

ID:	GMV UsageScenario 1	GMV UsageScenario 2	GMV UsageScenario 3	TEC UsageScenario 10
Related CaseStudy	CS4	CS4	CS4	CS4
Addressed Domains	Space	Space	Space	Space
Scenario Name	Component Reuse	Re-qualification	Safety analysis with AMASS platform	Model-Based Development
Short Description	This usage scenario aims to assess the feasibility of components reuse using different execution platforms	This usage scenario aims to analyse at model level, the impact of a re-qualification when the HW platform is modified	Analyse the system safety, performance, reliability and availability requirements using the AMASS platform	1) Support for Model-based System, Safety, and Security Co-Engineering 2) Support for Safety and Security Co-Analysis 3) Support for Safety and Security V&V 4) Support for Contract-based Design for System Architecture by Safety and Security Contracts 4) Architectural patterns: trade-off based on analysis and certification requirements 5) Fault Injection
Stakeholders	System Engineer Software Engineer Avionics Engineer Safety Engineer Test Engineer	System Engineer Safety Engineer Test Engineer	System Engineer Safety Engineer Test Engineer	System engineer Safety engineer Security engineer
Stakeholder constraints	None	None	None	None
Addressed Business Goals:	G1 G2	G1 G4	G1 G2 G3 G4	G4, G1, G3
Process Steps	· Requirements specification · Design · Model Analysis · Traceability	· Requirements specification · Design · Model Analysis · Traceability	· Requirements specification · Design · Model Analysis · Traceability	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification -System argumentation
Concerns	Safety	Safety	Safety	Safety, Security, Reliability, Availability, Maintainability
Cross-system certification	No	No	No	Yes
Cross-domain certification	No	No	No	No
Engineering Environment (Interoperability)	· Requirements and Desing: CHESS · Model Analysis (with tools integrated with CHESS, e.g., MAST, OCRA, xSAP, etc.). - Schedulability analysis. - Safety contracts. - FMEA/FTA.	· Requirements and Desing: CHESS · Model Analysis (with tools integrated with CHESS, e.g., MAST, OCRA, xSAP, etc.). - Schedulability analysis. - Safety contracts. - FMEA/FTA.	· Requirements and Desing: CHESS · Model Analysis (with tools integrated with CHESS, e.g., MAST, OCRA, xSAP, etc.). - Schedulability analysis. - Safety contracts. - FMEA/FTA.	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools-Safety/Security Analyses Tool and V&V Tools
Challenges	Improve the reuse based on software and hardware co-design.	Improve the re-qualification process.	Safety analysis at model design.	Safety and Security co-engineering, innovative fail-operational concepts, Safety and Security and Function Availability co-engineering, definition of architectural patterns
Standards	ECSS-M-ST-80C ECSS-Q-ST-10C ECSS-Q-ST-80C ECSS-E-70-41A ECSS-E-ST-10C ECSS-E-ST-40C ECSS-E-ST-30C ECSS-E-ST-40C	ECSS-M-ST-80C ECSS-Q-ST-10C ECSS-Q-ST-80C ECSS-E-70-41A ECSS-E-ST-10C ECSS-E-ST-40C ECSS-E-ST-30C ECSS-E-ST-40C	ECSS-M-ST-80C ECSS-Q-ST-10C ECSS-Q-ST-80C ECSS-E-70-41A ECSS-E-ST-10C ECSS-E-ST-40C ECSS-E-ST-30C ECSS-E-ST-40C	ECSS-Q-ST-30, ECSS-Q-ST-40, ECSS-Q-ST-80
Any wishes for usage scenario	-	-	-	Reuse of established safety methods for security topic
Any known constraints for usage scenario	-	-	-	Not so far

Table 16. CS4 TEC, FBK, INT and TAS-E usage scenarios

ID:	TEC UsageScenario 11	TEC UsageScenario 12	FBK UsageScenario CS4	INT UsageScenario 1	TAS-E UsageScenario 1
Related CaseStudy	CS4	CS4	CS4	CS4, CS10, CS11	CS4
Addressed Domains	Space	Space	Space	Space	Space
Scenario Name	Reuse of components	Assurance/Certification Management ToolAssurance/certification tool	CS4FBK	INT-US1	OLCI
Short Description	Reuse of components from one mission to another	Compliance with Standards/ product and process assurance/certification management tool to support the compliance assessment and certification	1) Specification of a subset of standard components used in space systems (including parameters and contracts) 2) Modeling of the system architecture using the library of components 3) Formalization of requirements and their refinement 4) Contract-based verification of the requirements refinement 5) Change architecture by changing decomposition and parameter values 6) Compare different architectures based on soft requirements and fault trees	Model-based System, Safety, and Security Engineering Support for Safety and Schedulability Analysis Contract-based Design for System Architecture by Safety and Security Contracts, Contract refinement formal verification	In-flight SW
Stakeholders	System engineer Safety engineer Security engineer Quality Assurance Manager	Quality Assurance Manager Safety engineer Security engineer	System engineer, Safety & Security engineer, ModelBased Safety researcher, Verification & validation researcher	System engineer Safety engineer Assurance engineer	Software Engineer
Stakeholder constraints	None	None	None		None
Addressed Business Goals:	G2	G4	G3	G1, G2, G3	G1, G2
Process Steps			system requirements system design system analysis system modeling system verification evidence for system argumentation	System requirements System design System verification System argumentation	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification -System argumentation
Concerns	Safety, Security, Reliability, Availability, Maintainability	Safety, Security, Reliability, Availability, Maintainability	SafetySecurityReliability	Safety	Reliability & safety
Cross-system certification	Yes	Yes			
Cross-domain certification	No	No	No		Space & Avionics
Engineering Environment (Interoperability)	Open Source tools AMASS tools when available for use and evaluation	Open Source tools AMASS tools when available for use and evaluation	CHESS interacting with analysis tools (OCRA, nuXmv, xSAP) and with OpenCert: Modeling in SySML or AADL using CHESS Formalization using CHESS/OCRA integration Validation and Refinement checked with OCRA Model checking with nuXmv FTA/FMEA with xSAP Collection of evidence for argumentation with OpenCert	CHESS and integration with analysis tools. CHESS tool supported activities are: Design, Dependability (usage of MDH tool) and Schedulability Analysis (usage of the MAST tool), Ada Code generation. Integration with OCRA and xSAP fro contracts verification and further dependability analysis support. Use of OpenCert AMASS environment to manage process, evidence, assurance case information.	• Architecture and Modelling: o Melody Advance o Microsoft Visio • Software Design: o Enterprise Architect o Rhapsody o Melody CCM • Software development: o Eclipse • Continuous integration: o Thales Control • Source control: o SVN o Git/Stash • Project and task management: o Jira • Requirements: o Doors
Challenges	Reuse of components targeting items of different SILs (&security levels)	argumentation patterns for fault tolerance, argumentation patterns for specific technologies	Reuse of components Comparison of architectures Application of formal methods Generation and reuse of evidence		
Standards	ECSS-Q-ST-30, ECSS-Q-ST-40, ECSS-Q-ST-80	ECSS-Q-ST-30, ECSS-Q-ST-40, ECSS-Q-ST-80		ECSS, SAVOIR-FAIRE	ECSS-E-ST-40C ECSS-Q-ST-80C
Any wishes for usage scenario	Reuse of established safety methods for security topic	Reuse of established safety methods for security topic	N/A	N/A	N/A
Any known constraints for usage scenario	Not so far	Not so far	N/A	N/A	N/A

2.5 CS5: Platform Screen Doors Controller

2.5.1 Short description of the case study

With the increase of the urban population in the world, more people need to be transported underground while existing infrastructure cannot be enlarged. The current solution to this problem, the reduction of the time interval between trains, requires adopting (semi-)automatic metros for the most crowded lines. However, these automatic trains have to stop at predefined positions on platforms in front of so-called platform screen doors, ensuring optimal passengers transfer between train and platform while avoiding passengers to fall on tracks at peak hours.

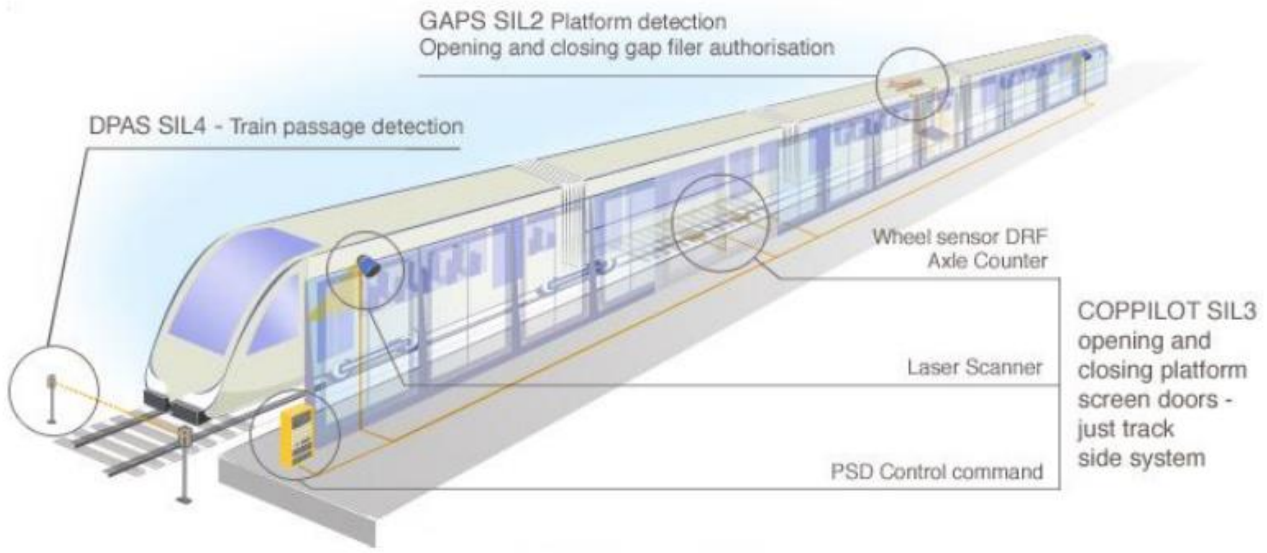


Figure 16. Example of combined systems to ensure safety at a metro platform

ClearSy develops safety critical systems that are often specified with a very concise requirement: “ensure a function at a safety level of {SIL2, SIL3 or SIL4} in less than xx milliseconds”. The systems engineering phase consists in refining this requirement into a set of functions that are distributed over an architecture that includes sensors, computers and actuators. Then the design phase and safety demonstration are performed in parallel in order to iteratively obtain a working, reliable and safe-enough system. System engineering is mainly based on human experience and expertise, Microsoft tools and sometimes on formal methods when some tricky aspects need to be managed or when trustworthy software is required. The combination of formal models of both discrete controllers and continuous physical environment helps to better analyse (some dimensions of) the system that could be animated/checked earlier. ClearSy develops both hardware and software of these systems in conformance with EN50126, 8 & 9 standards, including devices for fine-tuning sensors and supervision facilities. These systems have to provide safety functions that require cross-domain skills and knowledge, and dedicated/diverse engineering tooling.

2.5.2 Technical description of the case study

COPPILOT System controls PSD and is based on system detection. It includes a lot of sensors to be installed on the track side, and it doesn’t require any installation on board. It is a safety SIL3 system developed by ClearSy.

2.5.2.1 System description

To open the platform screen doors COPPILOT needs to know:

- If there is a train (F1)
- If the train is stopped (F2)

- If the train is at the right position (F3)
- If the train doors are opening (F4)

The opening control function is SIL3.

To close the platform screen doors COPPILOT needs to know:

- If the train doors are closing (F5)

The closing control function doesn’t need to be safe.

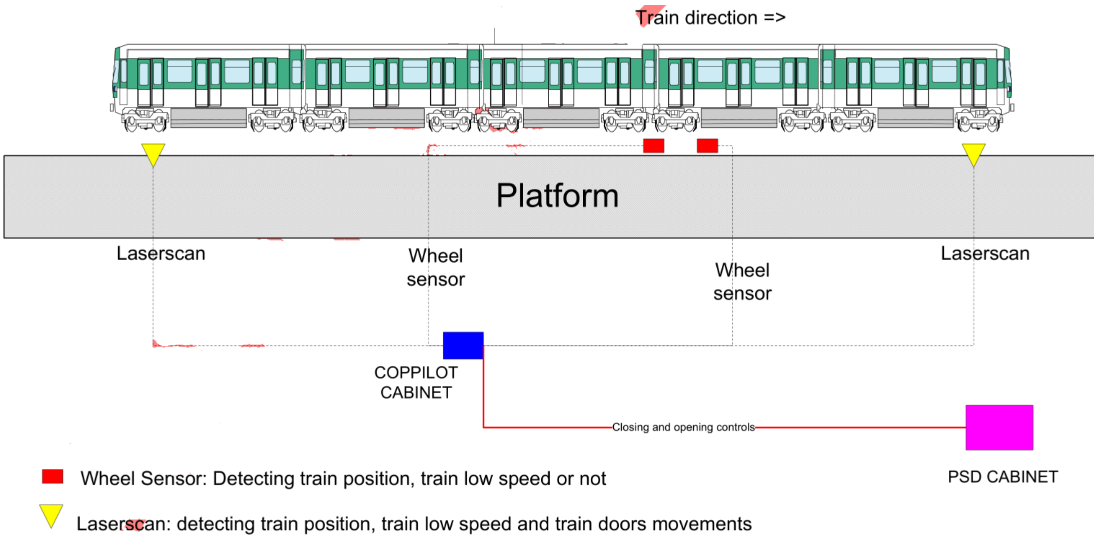

Train direction =>
Platform
Laserscan
Wheel sensor
Wheel sensor
Laserscan
COPPILOT CABINET
Closing and opening controls
PSD CABINET
🟥 Wheel Sensor: Detecting train position, train low speed or not
▽ Laserscan: detecting train position, train low speed and train doors movements

Figure 17. Platform screen doors COPPILOT

F1: If there is a train

COPPILOT must not open the PSD if the train is not stopped and is not in the right position range.

F2: If the train is stopped

COPPILOT must not open the PSD if the train is not stopped. The train is stopped if its speed is under 0,5km/h (speed limit fixed in COPPILOT).

Because COPPILOT has no inboard equipment, COPPILOT can’t stop the train. So, the train must be stopped at the right place when the driver or another system opens the PSD.

F3: If the train is in the right position

COPPILOT must open the door only if the train is in the right position range. This range is given by METRO (PAR). This range cannot exceed the length of an unprotected door by the train (SPAR).

F4: If the train doors are opening

When COPPILOT detects that the train doors are opening, if all three previous functions are true, then COPPILOT controls the opening of the PSD.

The duration of this action is 6 seconds maximum. COPPILOT controls the PSD opening until the opened PSD input is set or the 6 seconds delay has ended.

While the three functions are true COPPILOT can open the PSD with no limitation of time, each time it detects that the train doors are opening.

It’s not dangerous to open the PSD even if the train doors are not opened while the three functions are true, just because it’s not possible to fall in the tracks when there is a train stopped in the right position, in front of the PSD.

As it is not possible to make detection without delay, the PSD opening speed must be higher than the train doors opening speed if synchronisation of the doors is required. It’s almost true if the PSD lengths are higher of the train doors length, because the PSD have more distance to open.

F5: If the train doors are closing

When F1 and F2 and F3 are true, and if F4 occurs, COPPILOT is waiting for a train doors closing. When it detects this action then COPPILOT controls the PSD closure.

The duration of this action is 15 seconds maximum. COPPILOT controls the PSD closure until the closed PSD input is set or the 15 second delay is ending.

If during this closed action COPPILOT detects a train Doors opening, also COPPILOT stops the closing control and controls the PSD opening. As it is not possible to make detection without delay, the PSD closing speed must be higher than the train doors closing speed if synchronisation of the doors is required. It is almost true if the PSD length is higher than the train door length, because the PSD have more distance to close.

2.5.2.2 System architecture overview

COPPILOT SP is a safety system. The main function « open the PSD » is a SIL3 function.

Thus, the COPPILOT SP architecture is specified to match this requirement.

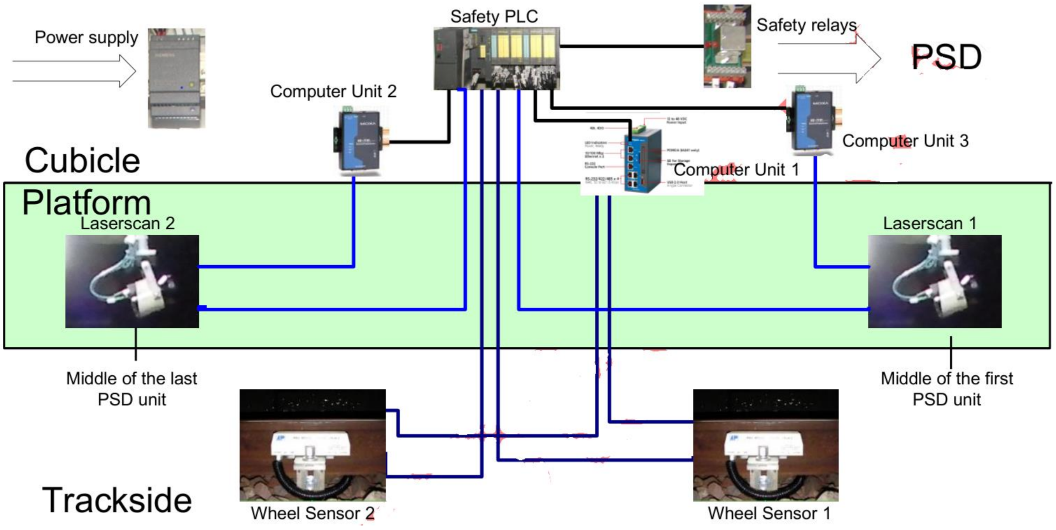


Figure 18. COPPILOT SP architecture

COPPILOT has no on-board element. COPPILOT uses sensors to detect all the events we need.

COPPILOT provides functions F1, F2, F3 and F4 by analysing the data given by each sensor. All these data are creating a “pattern”. This pattern is compared and validated to perform the functions in safety.

In case of two type trains or more, which include different distance between bogies or between cars, it would be possible that COPPILOT needs more than two-wheel sensors to validate that the train is inside the SPAR.

2.5.3 Case study state of the art

The platform screen door controller is a safety critical system. Its safety related requirements are in relation to EN50126, 50128 and 50129 standards. These standards describe at different levels what is expected for the target system, taking into account best practices accumulated over several decades.

2.5.4 Case study state of the practice

Development is made of two V cycles:

- Development cycle. A traditional process is used (system first, then hardware and software development in parallel). An early prototype, without any safety insurance, is built as a proof of concept for the target system. Traceability is ensured all along the process while requirements are kept as a low number (only main functions and safety requirements are selected). Development is based on PIC32 Microchip IDE and on Atelier B for the safety critical software. Documentation is made with Microsoft tools (Word, Visio, PowerPoint) while traceability is managed with Excel.
- Safety cycle. This cycle executes in parallel of the development cycle, by independent team. During this cycle, intense exchanges occur between the two teams, in order to consider all safety aspects early and to avoid deeply modifying the system later on. Analyses are made with Microsoft tools (Word, Excel, Visio).

2.5.4.1 Workflow

The safety team is in charge of ensuring a safety integrity level in line with the system requirement, as expressed by the final customer.

The main deliverable is the safety case. The document is divided in several sections:

- Hazard analysis, where risks are analyzed and safety functions are determined accordingly to their SIL level.
- Safety demonstration, where the proof of safety for the chosen safety function is provided at system level.
- Conclusion and exported constraints, where the global safety conclusions are started together with the necessary global exported constraints.

2.5.4.2 Assessment

Assessment is performed by a third party, namely CERTIFER French not-for-profit association.

This assessment is based on existing documentation that is provided during the lifetime of the project (and not all at once at its end). Several meetings are organized in order to share understanding and to address early possible.

2.5.4.3 Involved roles

Four different parties are involved:

- The design team in charge of developing and verifying the system, including the project manager, several software and electronic engineers.
- The validation team, made of several validation engineers.
- The safety team (one safety expert).
- The third-party assessor (ISA) from CERTIFER.

2.5.4.4 Tools and Tool chains

2.5.4.4.1 Used tools and methods (included guidelines)

Tools used are:

- Microsoft tools (Word, Excel, Visio, PowerPoint)
- PIC32 Microchip IDE
- Atelier B
- FIDES

2.5.4.4.2 Tool chain

Safety analysis is purely a text-based process. The resulting document, the safety case, can be considered as a thesis where the safety is demonstrated (regarding safety requirements) while considering a number of assumptions that are precisely described.

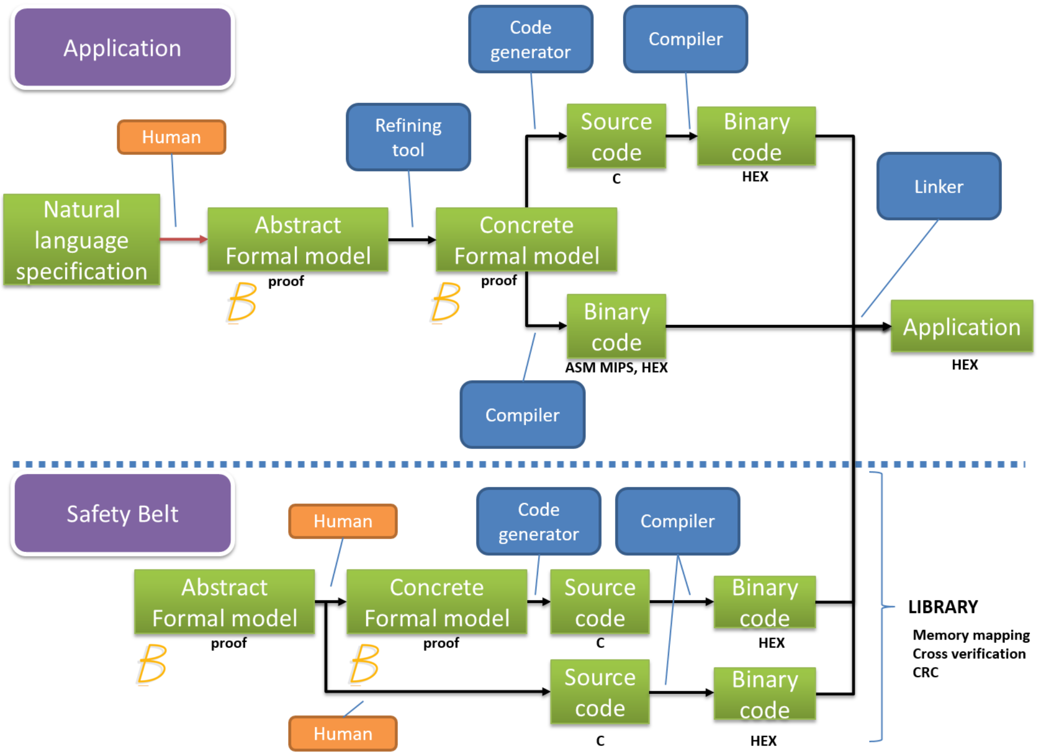


Figure 19. COPPILOT SP languages

Development is performed by using two different technical environments:

- Microchip IDE, for developing and compiling non-safety related software (C language),
- Atelier B, for safety-related software. Two different code generators are used:
  - A C code generator (C4B)
  - An Hex compiler, producing a binary file

Requirements management is performed with Excel. A strong focus is put on reducing the number of requirements to be considered in the development (something like 30 or 40 requirements for a system). The idea is not to be lost when managing hundreds or thousands of requirements but to focus on what is really important. Design documentation is cross-read, critical code reviews are performed on safety related software, requirements traceability is verified through tables (traceability, coverage) among / between design documents and testing documents.

There is no need to have a qualified tool, hence none of the ones used for the development are qualified. In the safety case, we have to make clear how possible mistakes are handled by the development / safety process.

2.5.5 Expected technical improvements

2.5.5.1 STO1. Architecture-driven Assurance

- N/A

2.5.5.2 STO2. Multi-concern Assurance

- Preliminary steps to integrate security concerns into safety concerns (not currently required by railways standards but an evolution on this aspect is forecasted in the coming years).

2.5.5.3 STO3. Seamless Interoperability

- Better level of confidence on the C code generation process that would ease / speed up critical code reviews (generation of assertions from B models to enable program proof with Frama-C).

2.5.5.4 STO4. Cross/Intra-Domain Reuse

- N/A

2.5.6 Business needs

2.5.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

Improving the code review process to lower verification costs and risks (better level of confidence in the software).

2.5.6.2 AMASS Goal 2

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*

N/A (improvements on code review are expected to be fully automatic and hence replay-able at will, so reusing previous assurance results is not particularly interesting).

2.5.6.3 AMASS Goal 3

> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

N/A

2.5.6.4 AMASS Goal 4

> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

N/A

2.5.7 Usage scenarios

Table 17 and Table 18 show the 6 usage scenarios related to Case Study 5.

Table 17. CS5 CLS and AIT usage scenarios

ID:	CLS UsageScenario 1	CLS UsageScenario 2	AIT UsageScenario 3
Related CaseStudy	CS5	CS5	CS5
Addressed Domains	Railway	Railway	Railway
Scenario Name	Proven-source-code	System-level-model	Automated train systems, platform doors
Short Description	Generation of Frama-C asserted C code from B models	Support for system-level model, including safety and security aspects, and test cases generation	system modelling and safety/reliability analyses attached to the system models, including system architecture, patterns of assurance and test case generation from system model
Stakeholders	Software developer Safety engineer	System engineer Safety engineer	System engineer Risk analyzer Safety engineer Security engineer Test engineer
Stakeholder constraints	None	None	None
Addressed Business Goals:	G1: to improve the level of confidence of the software application that runs on the execution platform (qualitative objective)	G1: to demonstrate the ability to fully analyze existing designs G2: to demonstrate the ability to perform both safety and security analyses	G1, G2, G4
Process Steps	Software design Software modelling Code generation	System design System analysis System verification (test) System modelling	Development and Assurance Process
Concerns	Safety	Safety and security	Safety, Security and Timing
Cross-system certification	N/A	N/A	No
Cross-domain certification	No	No	No
Engineering Environment (Interoperability)	Atelier B, Frama-C, Why3	Safety Architect	Tecnalia is not mentioned as participant in CS5
Challenges	N/A	N/A	includation of effect on timing from safety and security, during analysis and system design
Standards	EN50128	EN50129	EN50126/8/9 and new approaches (draft standard) towards usage of IEC62443 in railway domain
Any wishes for usage scenario	N/A	N/A	We should define used system models (modeling language and tools) as soon as possible
Any known constraints for usage scenario	N/A	N/A	Since communication based railway systems can be very complex, (multiple levels and trains) we would prefer if the scope could be clearly defined, e.g. one train at a station communicating with station

Table 18. CS5 FBK, A4T and CEA usage scenarios

ID:	FBK UsageScenario CS5	A4T UsageScenario 3	CEA UsageScenario 1
Related CaseStudy	CS5	CS5	CS5
Addressed Domains	Railway	Railways	Railway
Scenario Name	CS5FBK	Automated train systems, platform doors	
Short Description	1) Modeling of the system architecture and behavior 2) Formalization of the system requirements 3) Validation of the requirements 4) Model checking 5) Fault injection and FTA/FMEA	Model Based testing of Safety	Model-based System, Safety, and Security Engineering Support for Safety and Security Analysis Code-level verification
Stakeholders	System engineer, Safety & Security engineer, ModelBased Safety researcher, Verification & validation researcher	System engineer Safety engineer	System engineer, Safety & Security engineer, ModelBased Safety researcher, Verification & validation researcher
Stakeholder constraints	None	None	None
Addressed Business Goals:	G3	G1 - G2	G3,G4
Process Steps	system requirements system design system analysis system modeling system verification evidence for system argumentation	Product development on system level concerning safety -System requirements -System design -System verification -System argumentation	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling - code-level verification -System safety assurance case
Concerns	Safety Security Reliability	Safety	Safety Security Safety and Security
Cross-system certification	N/A	Between system components	N/A
Cross-domain certification	No	No	No
Engineering Environment (Interoperability)	CHESS interacting with analysis tools (OCRA, nuXmv, xSAP) and with OpenCert: Modeling in SySML or AADL using CHESS Formalization using CHESS/OCRA integration Validation and Refinement checked with OCRA Model checking with nuXmv FTA/FMEA with xSAP Collection of evidence for argumentation with OpenCert	MBSE + MaTeLo + Requirements engineering + Test benches	Papyrus and Sophia extension for safety/security analyses Frama-c for code level verification
Challenges	Application of formal methods Generation of formal proofs as evidence for the argumentation	Safety and test co-engineering - reusability	Safety and Security co engineering
Standards		EN 5012x	EN 5012x ISO 26262
Any wishes for usage scenario	N/A	N/A	Reuse of established safety methods for security topic and for co-assurance of safety and security
Any known constraints for usage scenario	N/A	N/A	N/A

2.6 CS6: Automatic Train Control Formal Verification

2.6.1 Short description of the case study

Alstom Signalling develops safety critical signalling systems for railway application (mass transit or main lines). These systems shall comply with international safety standards such as CENELEC EN50126/8/9, specific regional safety regulations and technical specification for interoperability (e.g. ERTMS specification in Europe).

The Alstom's signalling solutions portfolio contains several applications and technologies which comply with these safety standards. Each Alstom's signalling solution has its own safety demonstration based on Generic Application Safety Case (compliant with EN 50129 structure) that shall be instantiated by application engineering for each project. This instantiation releases a Specific Application Safety Case (SASC as per EN 50129 standard). Basically, the functional and engineering processes safety demonstrations are in the GASC while the specific application data and process assurance safety demonstrations are in the SASC. Both GASC and SASC shall be submitted to assessment bodies such as the Independent Safety Assessor (ISA).

The case study focuses on the application of formal method in safety demonstration of signalling systems. The goal is to industrialize safety properties modelling and capitalize processes, argumentation and artefacts of such kind of demonstration. A specific application of formal method at signalling system level will be provided to illustrate the proposed solutions.

2.6.2 Technical description of the case study

Some safety demonstrations stem from formal methods verification & validation activities. Formal method means in this context: strong mathematical foundation and precise without ambiguity. Formal Proof is verification or validation technique based on a formal method that determines the absence of errors. As examples of formal method applications for the Alstom Urbalis 400 Communication Based Train Control (CBTC), the Automatic Train Protection (ATP) is proven with the B-method, the system level application data are verified by means of constraint solver and interlocking (IXL) is verified with model checking techniques. All these applications of formal methods require the identification and modelling of safety properties that the system/subsystem shall fulfil. And all these applications of formal methods require to be assessed by certification bodies.

A CBTC system involves an Automatic Train Control subsystem (ATC) that involves itself two subsystems. A carborne controller (CC) installed on-board of each train made of an Automatic Train Operation (ATO) that controls the movement of the train and of the on-board part of the ATP that monitors the position and kinetics of the train and prevents it from exceeding its limits of movement and speed. And a zone controller (ZC), the part of the ATP installed in a technical room at trackside, which monitors the movements of all trains in the rail network, calculates their respective limits of movement and transmits them to the CCs.

Depending on the size of the railway network and the number of trains to be monitored, one or more ZCs may be installed. In the latter case, each of the ZCs monitors a particular zone of the network and exchanges information with the ZCs of the adjacent zones in order to ensure, as and when trains move from one zone to another, a correct, safe and continuous monitoring of train and calculation of limits of movement.

The case study will be the formal analysis of the communication between adjacent ZCs in order to demonstrate that in all circumstances and track configurations when a train is moving from a zone A to a zone B then the information exchanged between the ZCs in charge of these zones ensure that both ZCs know precisely (to the nearest error margin) the location of the train in the inter-ZC zones (i.e. the part of zone A bordering zone B that is known to the ZC of zone B and the part of zone B bordering zone A that is known to the ZC of zone A).

This case study intends to apply formal method on system or subsystem level functions to improve system specification or verify exhaustively the safe behaviour of the system/subsystem. Although these methods are now well known, they all shall identify the set of formal safety properties that the system, subsystem or data shall fulfil. Nowadays, there are no common methods to develop these formal safety properties. The primary objective of this case study is to develop a common method to derive formal safety properties of a signalling system/subsystem and to capture the refinement process and artefact into the certification framework. As far as possible, this method and the corresponding certification framework infrastructure shall be applicable to all kind of use of formal methods used for the verification or validation of safety properties within the development or deployment of a signalling solution.

This case study is under the responsibility of Alstom. AMASS partners may be involved in considering required data structures and method processes to be included in AMASS developments.

During the AMASS project the following points are planned to be performed within this railway case study:

- Methodology definition
- Data structures specification
- Safety analysis modelling
- Use Case application to a specific system proof
- Analysis of V&V and other cost reduction
- Integration of the proposed methodology within existing signalling solutions (inclusion in the Generic Application Safety Cases and ISA strategy).

2.6.3 Case study state of the art

CENELEC standards EN50126, EN50128 and EN50129 provide a set of requirements on processes, organization and techniques with which the development, deployment and maintenance of the railway signalling systems shall comply.

Standard EN50126 provides a process which enables the implementation of a consistent approach to the management of reliability, availability, maintainability and safety (denoted by the acronym RAMS) of total railway systems (including but not restricted to signalling). This Standard:

- Defines RAMS in terms of reliability, availability, maintainability and safety and their interaction;
- Defines a process, based on the system life-cycle and tasks within it, for managing RAMS;
- Enables conflicts between RAMS elements to be controlled and managed effectively;
- Defines a systematic process for specifying requirements for RAMS and demonstrating that these requirements are achieved.

Standard EN50128 provides a set of requirements with which the development, deployment and maintenance of any safety-related software intended for railway control and protection applications shall comply. It defines requirements concerning organizational structure, the relationship between organizations and the division of responsibility involved in the development, deployment and maintenance activities. Criteria for the qualification and expertise of personnel are also provided in this Standard.

Standard EN50129 provides a set of requirements that shall be satisfied in order that a safety critical railway system/subsystem/equipment can be accepted as adequately safe for its envisioned application.

The documentary evidence that these requirements 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.

Alstom's case study for AMASS project shall comply, for its concerned parts, with these standards.

Many case studies and applications of formal methods have been carried out in the past ([26],[27],[28]). Of all of them, the most relevant and inspiring for our case study is [28]. This is a safety analysis that demonstrated that the “no collision” safety property of a complete railway signalling system is ensured based on the safety properties that each of the sub-systems of this system ensures. The analysis used formal models of the relevant parts of the complete system and subsystems and a significant part of the demonstration was brought by formal proof. Formal modelling was done with Event-B ([29]) and the proof was done with Atelier B.

In Alstom’s case study we shall follow a similar approach and we shall use the same technology.

2.6.4 Case study state of the practice

The figure below gives a simplified representation of Alstom’s CBTC development life-cycle.

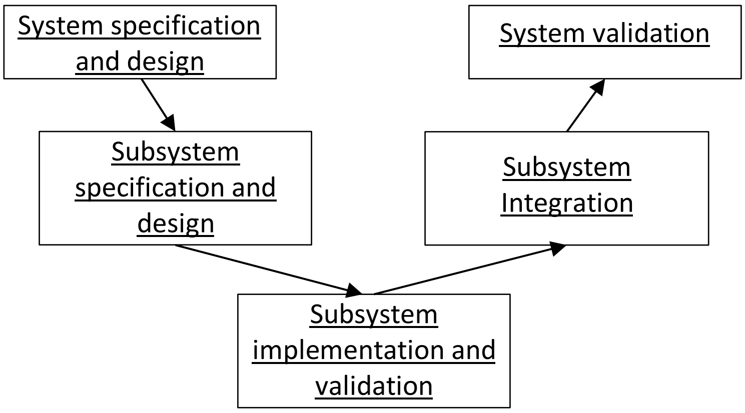

System specification and design
System validation
Subsystem specification and design
Subsystem Integration
Subsystem implementation and validation

Figure 20. Alstom’s CBTC development life-cycle

The purpose of system specification and design phase is to define CBTC’s system requirements and architecture, and to allocate requirements on system components (subsystems). Functional decomposition and semi-formal structured methods are employed for this phase.

The subsystem specification and design phase has the same purpose and employs the same methods as the previous phase but at subsystem level. It is during this phase that ATC subsystem is specified and designed.

The purpose of subsystem implementation and validation phase is to define the requirements on the hardware and software products that implement the subsystems, to implement and test those products and to validate those products according to their requirements.

As written above, Alstom employs formal methods for the development and/or verification of some of Urbalis 400’s subsystems, i.e. the B Method ([23]) and the Atelier B tool ([24]) are used to design, implement and proof the software of on-board and trackside parts of Urbalis 400’s ATP, and HLL and the S3 tool ([25]) are used to verify formally that Urbalis 400’s IXL complies with safety requirements.

So, regarding our case study, presently we can demonstrate that the software of ZC complies with its formal specification but, as there is not a formal analysis at system or subsystem levels there is not a formal model of communicating ZCs and we cannot demonstrate formally that the communication of adjacent ZCs ensures both safe monitoring of trains and safe calculation of limits of movement.

2.6.4.1 Workflow

The objective of the ALSTOM case study is to create a safety assurance project for an Automatic Train Control signalling system that includes formal proof demonstration for an Automatic Train Control railway signalling system. The aim of the formal proof is to increase the early discovering of bugs and the general safety of the project. It can also replace a part of the functional tests, as all functional tests related to the proved property can be avoided, **if and only if** it is proved that the implementation of the system specification (Sw) is consistent with the formal model that has been proved.

As per any Signalling Railway safety demonstration, several standards are generally applicable. In Europe and, usually for international projects, the use of EN 50126, EN 50128 and EN 50129 are mandatory.

The safety assurance project shall include all artefacts required by the EN 50129 Generic Application Safety Case. These artefacts could be a reference to a document, table, diagram or text.

The application of the EN 50129 requires independence between the designer, the verifier (V&V) and the safety validation team.

The overall workflow of the system development is shown in Figure 21. The objective of the case study is to manage the safety evidences required by the standards for this specific application.

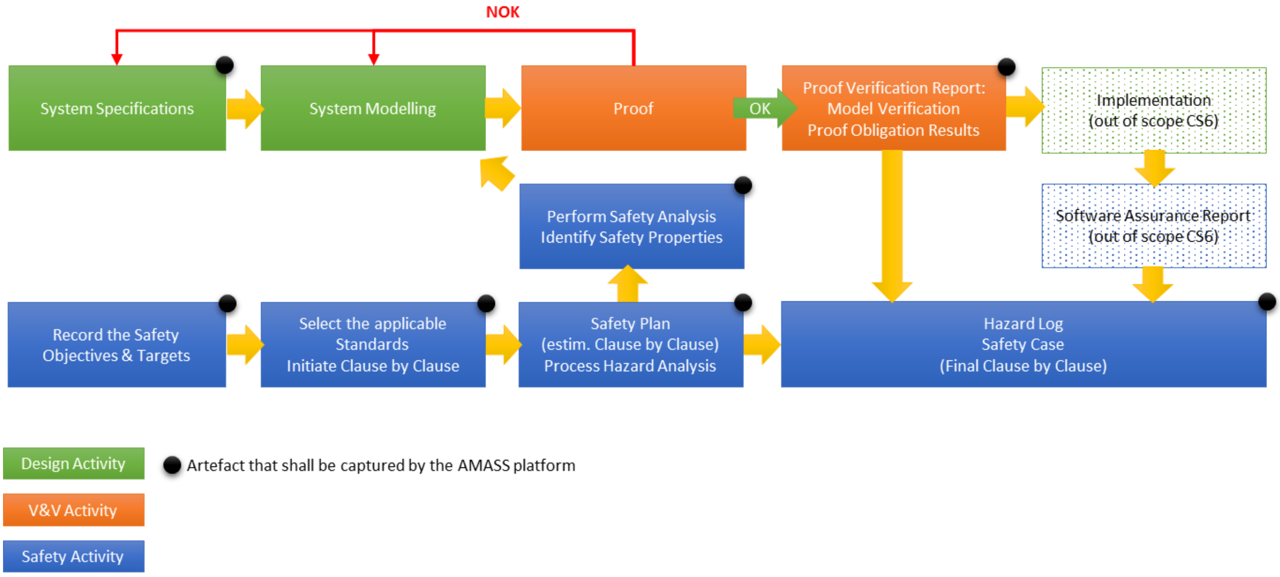

NOK
System Specifications
System Modelling
Proof
OK
Proof Verification Report: Model Verification Proof Obligation Results
Implementation (out of scope CS6)
Software Assurance Report (out of scope CS6)
Perform Safety Analysis Identify Safety Properties
Record the Safety Objectives & Targets
Select the applicable Standards Initiate Clause by Clause
Safety Plan (estim. Clause by Clause) Process Hazard Analysis
Hazard Log Safety Case (Final Clause by Clause)
Design Activity
V&V Activity
Safety Activity
● Artefact that shall be captured by the AMASS platform

Figure 21. CS6 Workflow

The Design actors propose system specifications (system requirements specifications of ATC and ZC, System Interface Description between two adjacent ZC). Then they model formally these specifications. In parallel, the safety actors identify the safety objectives and targets for the system, select the appropriate standards and initiate a clause by clause table. The second activity of the safety actors is to prepare the strategy of demonstration in the Safety Plan. At this step, they answer the standard clause by clause analysis with estimated evidence (not already performed but planned). Considering the system specifications, the safety engineers perform system hazard analysis and identified the safety properties that the system shall fulfil. These properties are integrated within the system formal model (either by safety actor or design actor). These last two activities may require several iterations.

When the model is ready, the V&V actors have two main activities to perform:

- The verification of the model: this activity shall demonstrate that the formal model is consistent with the system specifications, and, that the safety properties are consistent with the system hazard analysis;
- The proof: by means of formal proof assistant, the V&V actors perform the proof of the model. There are two possible outputs:
  - The proof fails, one or several counter-example disclose the safety properties: the system specifications or the system formal model shall be reworked (and, the verification of the model);
  - The proof succeeds.

At this step, the safety assurance project has demonstrated that the system specifications are safe. The design team can start the implementation of the system (the safe implementation evidence is out of the scope of this case study).

The safety assurance manager prepares the Hazard Log and the safety case with the evidences provided by the proof. He performs the final clause by clause analysis and provides for each safety property the result of the related proof obligation(s).

2.6.4.2 Assessment

The safety assessment is performed by an independent safety assessor. It consists in the evaluation of all the documents delivered for the system baseline (Design, V&V and Safety). A safety assessment report and a certificate of type (conformance with EN 50129 and EN 50128) are produced.

A safety assessment is an analysis to form a judgement, based on evidence that the System meets the specified safety requirements. The safety assessment addresses:

- The correct application of the safety management process (confirming that the activities are done in compliance with the defined requirements and process, including suitability of Safety Plan and related safety organization with respect to EN50126, EN50129, EN50128).
- The results of this application: checking that the results of the application of the above-mentioned process reaches the planned safety objectives of the new or modified system and checking that relevant limits of safety use are clearly and unambiguously expressed (i.e. completeness of safety-related application conditions, including restrictions and mitigations if necessary).

The assessor performs a review of all assurance project documents and transitively of all referenced documents. The assessor verifies the conformance of evidences provided for the EN 50129 and EN 50128.

2.6.4.3 Involved roles

The roles involved in this Case Study follows the prescription of the EN 50129 standards for SIL3 or SIL4 development (see Figure 22):

- Design actors (DI in 50129):
  - Write a specification of the system and its interfaces.
  - Model the specification within its environment in formal language.
  - *After the verification of the specification by formal proof, the designers implement the system (but this is out of scope of this Case Study).*
- V&V actors (VER in 50129):
  - Perform model verification (including traceability between the specifications and the model).
  - Perform the formal proof activity (if counter examples are found, the system specification. or the system model shall be updated, if the proof is set it authorizes the implementation of the specification).
- Safety actors (VAL in 50129):
  - Records the safety objectives & targets of the system.
  - Identify the applicable standards (we will consider in the CS6, the EN 50128:2011 and EN 50129 only), and initiate a clause by clause table (only standards requirement capture).
  - Write the de safety plan (organization, safety activities, fulfill the estimated clause by clause evidences).
  - Perform the process hazard analysis of the validation process of the system specifications (*i.e.* conformity with the EN 50128:2011 tool classes).
  - Perform the safety analysis of the system specifications which leads to the identification and the modelling of the safety properties to prove.
  - Write the Hazard-Log and the safety case that includes the evidences that the system specification is safe.
- Independent Safety Assessor (ASSER):

- Assess the safety assurance project
- Write the safety assessment report
- Deliver a certificate of type for the developed system

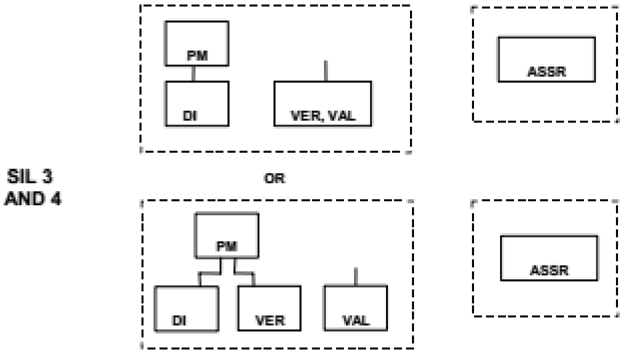


Figure 22. EN 50129 organization

2.6.4.4 Tools and Tool chains

2.6.4.4.1 Used tools and methods (included guidelines)

For formal modelling, we shall use Event-B and the Atelier B toolset.

Event-B is a formal language that allows users to model systems in terms of a state made of variables and a set of triggerable events that modify the value of the variables and proof that the events preserve the characteristic properties of the variables. It is possible to decompose Event-B systems in subsystems and proof that the decomposition preserves the characteristic properties of the decomposed system.

Atelier B provides the tools that check the syntax and the types the models written in Event-B language; that generates the proof obligations created by Event-B models; and that discharge these proof obligations.

2.6.4.4.2 Tool chain

Not Applicable.

2.6.5 Expected technical improvements

This CS will benchmark:

- Seamless link to system modelling (behaviour, safety, timing, signalling parameters).
- Methodological support and guidelines.
- Compliance management for safety standards for the Railway Domain.
- Tool support for impact analysis (safety analyses what ifs, etc.).
- Tool development to enhance, improve and enable the assurance case generation (safety).
- Arguments/Evidences reuse.

2.6.5.1 STO1. Architecture-driven Assurance

The benefits provided by this case study are to provide to the project team a systematic workflow for the safety assurance process. By means of the workflow assistant and the status of the achievements, some of the quality evidences are automatically recorded.

2.6.5.2 STO2. Multi-concern Assurance

Not applicable.

2.6.5.3 STO3. Seamless Interoperability

The AMASS platform will provide support for better interaction between each actor involved in the safety assurance project. It also insures by means of credentials setting the independence between the categories of actors as per standard requirements prescriptions.

2.6.5.4 STO4. Cross/Intra-Domain Reuse

Several things may be reused:

- EN 50129 and EN 50128 Clause by Clause (empty or estimated), as compliance management reuse
- Workflow of safety assurance project related to system level formal proof evidence
- Safety Plan Template
- Safety Case Template
- Hazard Log Template
- Safety Assurance Project Structure.

2.6.6 Business needs

2.6.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

The case study helps the adoption of formal method techniques at system level safety demonstration. The AMASS platform provides support for EN 50129, EN 50128 standards compliance allowing designers, verifiers and safety assurance managers to concentrate their effort on their best added values.

The adoption of formal method at system level would reduce considerably the late discovering of bugs in the final phases of the V-Lifecycle. The experiences of classical workbench test shows that lot of bugs are discovered in the ascendant phase of the V-Lifecycle. This leads to very costly rework of the system. The adoption of formal method at system level in early design phase of the development allows 30% effort reduction.

The standards compliance management is very costly when performed manually. The Clause by Clause is usually initiated for each new project; the workflow deviation may lead to several reworks (due to process owner audit and verification). The methodological support helps better understanding of the project process (workflow) and compliance requirements (clause by clause artefacts) and seamless interoperability leads to 20% effort reduction.

2.6.6.2 AMASS Goal 2

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*

Reuse is a key improvement in this case study. The formal method technique argumentation to comply with the EN 50129 and EN 50128 could be easily reused. The clause by clause analysis could also be fully reused for all projects using or not the formal method technique at system level.

2.6.6.3 AMASS Goal 3

> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

The raise of technology innovation is improved mainly by adoption of early validation techniques of system development. It allows better understanding of system need and environment enacting new way of thinking.

2.6.6.4 AMASS Goal 4

> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

This case study will develop a complete safety assurance project for an argumentation pattern at system level that could be standardized at company level and company and independent safety assessor level.

2.6.7 Usage scenarios

Table 19 shows the 4 usage scenarios related to Case Study 6.

Table 19. CS6 ALS usage scenarios

ID:	ALS UsageScenario 1	ALS UsageScenario 2	ALS UsageScenario 3	ALS UsageScenario 4
Related CaseStudy:	CS6	CS6	CS6	CS6
Addressed Domains:	Railway	Railway	Railway	Railway
Scenario Name:	Assurance Project Creation	System Design, V&V and Dependability Assessment	Evidence Management	Compliance Management
Short Description:	This activity is related to the creation and the setting of the assurance project in the AMASS platform. The output shall be: • Creation of the roles and definition of credentials (implementation of the independence between Design, V&V and Safety roles). • Workflow definition and allocation: define the stream of activities and allocate activities to actors. • Creation of the assurance project artefact structure: reference to a document, table, text, diagram, etc. • EN 50129 and EN 50128 clauses hierarchically captured (Hypothesis: Standards are recorded in the AMASS platform within a library for reuse purpose).	This is the basic usage; the actors follow the process with the workflow assistant and provide their baseline artefacts when necessary.	Description Recording and retrieving of consistent artefacts for a baseline system specification.	The compliance management is performed by Safety Assurance manager directly within the EN 50128 and EN 50129 tables. For each clause, the Safety Assurance manager provides a justification (not applicable because …) or an artefact of one baseline process assurance project (reference to a document, table, text or diagram). For each clause of the standards provide an estimated response or the realized response.
Stakeholders	AMASS Platform Administrator – Safety Assurance Manager – Design leader, V&V Leader.	Safety Assurance Manager – Design leader, V&V Leader.	Safety Assurance Manager – Design leader, V&V Leader.	Safety Assurance Manager
Stakeholder constraints	The credentials shall insure EN 50129 independence between category of actors. The standards clauses are uploaded in the project and shared with all actors.	The credentials shall insure EN 50129 independence between category of actors. The standards clauses are uploaded in the project and shared with all actors.	The credentials shall insure EN 50129 independence between category of actors. The standards clauses are uploaded in the project and shared with all actors.	N/A
Addressed Business Goals:	G1 ; G2 ; G3 ; G4	G1 ; G2 ; G3 ; G4	G1 ; G2 ; G3 ; G4	G1 ; G2 ; G3 ; G4
Process Steps:	• Assurance project creation. • General information about the project: Name, schedule (T0, Tend, date of safety case delivery, customer, independent safety assessor name). • Project actor information: o Name, Telephone, email o Category: Designer, V&V or Safety • Workflow: setting the activities, their interdependency, allocation to actors. • Artefact structure creation (reference to a document, table, text or diagram). • Credential settings: regarding the category of each actor, allow read/write credential to each project object within the AMASS platform. • Upload EN 50128 and EN 50129 clauses from library.	1. If needed, create a new baseline: the Design leader provides the new system specifications, the V&V Leader and the Safety Assurance Manager perform their impact analysis [the impact analysis may be automatized if artefacts are traced each other]. 2. Each actor provides his artefacts at each step of the safety assurance process.	The actors shall be able to store their artefact and their links with requirements of the safety assurance project. Configuration management shall allow actors to work on local or working copy of these artefacts before freezing them into a system baseline. The actors shall be able to request the AMASS platform for retrieving a specific artefact (included in the project structure) related to a system specifications baseline.	There are two steps in the CS6 process to perform compliance: • During the Safety Plan redaction: to perform estimated standards compliance (the plan to reach the compliance). • During the safety case redaction: to perform the resulted standards compliance (how the project reach the compliance). For each clause, the safety assurance manager provides an artefact from the on-going baseline of the safety assurance project (reference to a document, table, text or diagram).
Concerns:	Safety	Safety	Safety	Safety
Cross-system certification:	Automatic Train Control	Automatic Train Control	Automatic Train Control	Automatic Train Control
Cross-domain certification:	no	no	no	no
Engineering Environment (Interoperability):	MBSE for artefact recording and argument structure	Formal Method	MBSE and Formal Method	MBSE and Formal Method
Challenges :	early validation	early validation & formal method argumentation at system level for 50129 compliance	reuse of early validation & formal method arguments	Accurate compliance management. Facilitate safety assessment acceptance.
Standards:	EN 50128 et EN 50129	EN 50128 et EN 50129	EN 50128 et EN 50129	EN 50128 et EN 50129
Any wishes for usage scenario	N/A	To manage configuration management (ie. baseline of the studied system)	N/A	To manage configuration management (ie. baseline of the studied system)
Any known constraints for usage scenario	N/A	N/A	N/A	N/A

2.7 CS7: Safety Assessment of Multi-Modal Interactions in Cockpits

2.7.1 Short Description of the Case Study

The current market indicates significant push to decrease the pilot workload and to further increase the operational safety. With the advent of human machine interface (HMI) beyond legacy displays, new safety assessment methods are required which both handle increased complexity and at the same time allow leveraging truly independent means of cockpit/pilot communication. The trend is to integrate touch screens, speech recognition systems and gaze tracking systems into new cockpit generation. Even though these technologies are mature enough in consumer markets, they have not found its way into avionics due to unresolved safety constraints. The suggested case study aims at handling the safety case in a progressive and flexible manner and alleviating from all potential safety hazards. This case study will be driven by Honeywell (HON). Masaryk University (UOM) and Fondazione Bruno Kessler (FBK) will contribute by providing methods and tools for automatic formal verification and validation and formal safety analysis. TEC will contribute to this case study by providing its experience from the OPENCOSS case study for avionics.

IFX and LAN will join in the aerospace case study with focus on aviation. The criticality class for aviation (class or Design Assurance Level – DAL C and D) may be compatible with the ASIL class B for automotive vehicles where Infineon will use its experience in automotive safety to exploit it in aerospace aviation. LAN will conduct research on the energy supply (APU – Auxiliary Power Units), the board network of aircrafts and their interaction with a central computing system within the Aviation Design Assurance Level DAL C and D. RPT will provide support for ensuring performance requirements can be met in practice.

2.7.2 Technical Description of the Case Study

When electronic displays are used as reconfigurable multifunction controls, the unique Human Machine Interface (HMI) is created allowing functions not possible with only traditional dedicated knobs, controls or physical hardware. Multifunction Controls could be cursor control devices, menu-based controls, touch screens, voice recognition and voice activated controls.

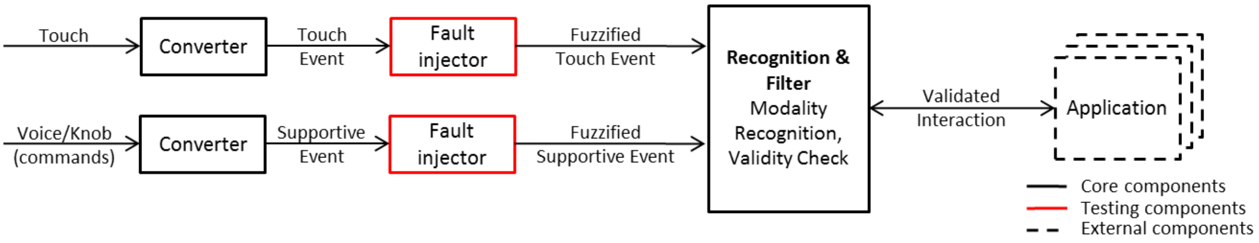


Figure 23. Multifunction controls

Touch screens are based on different technologies – acoustic, capacitive, infrared, resistive or strain gauge. Touch screen contains touch panel for crew input using gestures or touches with the touch targets displayed on the Liquid Crystal Display (LCD). Each target is defined as an active area of the touch panel. When such area is contacted by cockpit crew it initiates an action. The LCD also displays aircraft system information to provide the crew with information that can be used to guide control actions or to provide situational awareness.

Filter component combines inputs from multiple modalities and produces high-integrity interaction output. This way some failures, for example unintentional touch, are detected and ignored. The faults can be injected automatically to enable verification of the system. The recognizer component recognizes touch events and individual gestures.

This Case Study 7 will focus mainly on the following 3 Usage Scenarios:

- US1: Application of aerospace industrial standards for safety assessments
- US2: Automation of the verification objectives
- US3: Reuse of assurance artefacts from automotive technology into the avionics domains

US1 and US2 will consist of the following activities:

- Case study requirements (safety, performance, behavioural) will be captured and formalized.
- The formalized requirements will be automatically analysed including change impact analysis. Verification whether high-level requirements are covered by low-level requirements will be performed. Automatic detection of partially redundant and inconsistent formalized high-level requirements will be deployed.
- High-level system will be modelled and the corresponding (i.e. high-level) requirements will be formally verified against the low-level requirements written in Simulink or C system design using model checking and testing.
- Safety and dependability analyses of the multi modal interactions will be performed in order to generate Fault Tree analysis and FMEA. Some of these activities will be automated using AMASS tools.

The US3 will be performed independently by IFX and LAN and will concentrate on the reuse of certification artefacts of automotive components in the aviation domain. IFX semiconductor components like microcontrollers, energy supply circuits and highly efficient switching parts are used in huge amounts in the automotive domain. As a result, high reliability is proven in practice in a mass market in opposite to the aviation domain with small or medium quantities of units. The advantages of having such experience by using certification aspects of these parts in the aviation domain are obvious in respect of safety and reliability. Therefore, IFX and LAN will research if a cross domain reuse of evidence and artefacts from automotive assurance processes can be assisted by using AMASS tools and methods. IFX will analyse which models, documents and other kind of artefacts are available and transfer them together with LAN for usage in the AMASS environment. The intention of LAN is to use this work for further certifying activities of a new small aircraft with electrical propulsion.

Furthermore, LAN will investigate during the project how the AMASS tool platform supports the usage of automotive components and thereon based development for aviation purpose in terms of evidence management, traceability and seamless integration.

2.7.3 Case Study State of the Art

The safety assessment is done in conformance with the guidelines provided in ARP4761, Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment, Honeywell Product-Level Safety Assessment Work Instruction document, Federal Aviation Administration (FAA) regulations, and European Aviation Safety Agency (EASA) regulations. The EASA is the successor of the European Joint Aviation Authorities (JAA) and JAA is still used in ARP4761. Some EASA regulations are inconsistent with FAA regulations or in addition to FAA regulations.

Issues beyond state of the art:

- Multifunction controls bring new challenges about how to prevent inadvertent operations.
- Multifunction control may not have its location fixed and may need to be located by navigating through various menus or pages. When a control activates multiple different functions based on gestures or selections each function shall be distinctively marked.
- Reliability and formal verification of gesture recognition, conflicting gestures avoidance and reliable recognition of a set of gestures is a challenge.
- Open question is how to describe the failure model for the system and for the gesture recognition.

- Multi-touch technology requires engineers to write the code for filtering, recognizing, and matching of custom library of gestures. The challenge is to reliably determine the complete set of gestures and to determine if the new gesture conflicts with the existing ones. Possible solution is to use regular expressions to represent multi-touch gestures as suggested in paper [10].

2.7.4 Case study state of the practice

In aerospace domain, FAA requires that all electronic hardware in airborne systems qualify to RTCA DO-160 specifications.

The safety assessment is based on a functional assessment that includes development of safety assumptions and mitigations, identification of hazards applicable to the multi modal interactions in cockpit, development of safety requirements based on the identified hazards, Fault Tree Analysis (FTA), Failure Modes and Effects Analyses (FMEA), Common Mode Analysis (CMA), Design Assurance analysis, Single Event Effects (SEE) analysis, and partitioning analysis. The plan is to leverage of Models-Based Safety Assessment (MBSA) annex to the ARP 4761.

Current state of the practice, at least at Honeywell, is that these artefacts are developed mostly manually using Microsoft Office.

2.7.4.1 Workflow

The workflow for AMASS related aspects of the case study is under development and is captured in the documents [11], [12], [13] and [14].

Other development activities are summarized in the following tables. Table 20 summarizes the current state of the practice in Honeywell, the baseline of the activities and tools that were used for development of the case study before AMASS. The second Table 21 lists activities and tools that will be used during AMASS to demonstrate AMASS platform benefits. The metrics from baseline development process will be compared to metrics gathered from AMASS development process for evaluation of the AMASS contributions.

Table 20. CS7 Workflow – baseline – tools used for development before AMASS

DEVELOPMENT PHASE	ACTIVITY	TOOLS (Initial proposal) of the AMASS Reference Architecture
Requirements Specification & Architectural Design	Requirement Authoring	MS Word, DOORS
	Requirement Formalization	Not performed at all
	Traceability	Excel spreadsheet, references
Detailed Design and Code Generation	Model-based design and code generation	Matlab/Simulink, HAM
Validation & Verification	Consistency, redundancy & vacuity checking	Manual reviews
	Model checking	Not performed at all
	Test generation	HiLiTE
	Safety Assessment	MS Word

Table 21. CS7 Workflow – expected tools to be used for development thanks to AMASS

DEVELOPMENT PHASE	ACTIVITY	TOOLS (Initial proposal) of the AMASS Reference Architecture
Requirements Specification & Architectural Design	Requirement Authoring	MS Word, DOORS, Property Manager
	Requirement Formalization	Property Manager

	Traceability	Excel spreadsheet, references, Evidence Manager
Detailed Design and Code Generation	Model-based design and code generation	Matlab/Simulink, HAM
Validation & Verification	Consistency, redundancy & vacuity checking	V&V Manager and Verification Servers
	Model checking	V&V Manager and Verification Servers
	Test generation	V&V Manager or HiLiTE
	Safety Assessment	MS Word or CHESS

Traceability links shall represent the bridges between various related artefacts, inevitably also between the developed analysis/design/code on one side and the assurance related products on the other.

2.7.4.2 Assessment

Display and Graphics centre of excellence (independent Honeywell department) will benefit from the formal verification and safety assessment results and will evaluate the benefits of AMASS results. The return in investment analysis will be performed to assess whether it makes sense to deploy proposed technology.

2.7.4.3 Involved roles

The following roles will be involved:

- System Engineer
- Verification and Validation Engineer
- Safety Engineer
- Avionics Engineer (LAN only)
- Quality Assurance Manager (TEC only)

2.7.4.4 Tools and Tool chains

2.7.4.4.1 Used tools and methods (included guidelines)

These standards contain the relevant guidelines: RTCA DO-178C, ARP 4761, IEC 61508, RTCA DO-254, RTCA DO-278A, and SAE-ARP 4754.

The tools listed in the Table 22 are employed in the Case Study 7.

Table 22. Tools used in Case Study 7

Tool	Functionality	Interoperates with tools
MS Word	Requirements authoring, safety assessment	
MATLAB/Simulink, HAM	Architecture definition, design, automatic code generation	
JIRA	Problem reporting	
SVN	Configuration management	
HiLiTE	Test vector generation	MATLAB/Simulink, HAM
Property Manager	Requirement authoring and requirement formalization	
V&V Manager	Consistency, redundancy & vacuity checking, Model checking, Safety Assessment	

When Honeywell Auto-code Manager (HAM) library is used in Simulink, Honeywell HiLiTE tool is qualified for selected programs and to be used for design assurance level A and Tool Qualification Level 4 (TQL-4) for automated test generation and test harness.

2.7.4.4.2 Tool Chain

AMASS tools and methods will be applied during the whole process as appropriate. There is currently no usage of the Eclipse platform in the original tool chain, whose constituent parts are presented in the Table 22. Therefore, it is necessary to define the mapping of the AMASS-unaware development artefacts and procedures to the elements provided by the AMASS project and streamline the development through both (so far separate) platforms.

2.7.5 Expected technical improvements

Perform assessment of STO1 (System Architecture-driven Assurance), STO2 (Multi-concern assurance), and STO4 (Cross-Domain and Intra-Domain Reuse). More concretely, this CS will benchmark:

- Seamless link to System Modelling (Behaviour, Safety, Timing, etc.).
- Tool support for formal verification and analysis (including impact analysis).
- Formal safety analysis tools and methods – leverage of Models-Based Safety Assessment (MBSA) annex to the ARP 4761, which is the safety assessment guideline for aerospace.
- Cross-domain reuse activities supported by the AMASS platform.
- Methodological support and guidelines.

Masaryk University (UOM) and Fondazione Bruno Kessler (FBK) will contribute by providing methods and tools for automatic formal verification and validation and formal safety analysis.

COMPASS Tool set was previously evaluated by Honeywell.

Table 23. Proposed tool chain to be deployed for Case Study 7

Tool	Functionality	Interoperates with tools
MS Word	Requirements authoring	ForReq
ForReq	Requirements authoring, formalization, and formal verification	MS Word, MATLAB/Simulink, HAM, DIVINE, NuSMV, nuXmv
MATLAB/Simulink, HAM	Architecture definition, design, automatic code generation	ForReq
DIVINE	Model checking	ForReq
nuXmv, NuSMV	Model checking	ForReq
OCRA, xSAP	Safety assessment	
Acacia	Realizability checking	ForReq
JIRA	Problem reporting	
SVN	Configuration management	

2.7.5.1 STO1. Architecture-driven Assurance

- Extension of the SysML modelling of safety and system architecture (Enterprise Architect). Honeywell 3 View System Engineering.

2.7.5.2 STO2. Multi-concern Assurance

- Automated Assurance – Semantic Verification of the Formal requirements.
- Verification and Validation of behavioral formal requirements and safety requirements against the system architecture and the system design.

2.7.5.3 STO3. Seamless Interoperability

- Seamlessly integrated tool chain to automate formal verification and safety assessment. Based on OSLC.
- Evaluate benefits of ModelBus.

2.7.5.4 STO4. Cross/Intra-Domain Reuse

- Reuse of the existing artifacts within aerospace domain.

2.7.6 Business needs

2.7.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

- Methodology and SysML modelling tools to perform safety assessment.
- Methodology of seamless connection between the different tools (e.g. ForReq, Simulink, SysML modelling tools, verification and safety assessment tools).
- Methodology and tools to semantically analyse the requirements.
- Methodology and tools to formally verify the requirements against system architecture and system design.
- Methodology and tools to perform safety assessment.

2.7.6.2 AMASS Goal 2

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*

- Automated safety assessment results reuse.
- Reuse of formal verification results.
- Safety assessment argumentation methods.

2.7.6.3 AMASS Goal 3

> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

- Methodology and tools to semantically analyse the requirements and reduce the propagated defects.
- Methodology and tools to formally verify the requirements against system architecture and system design and to reduce propagated defects.
- Methodology and tools to perform safety assessment at least semi-automatically.

2.7.6.4 AMASS Goal 4

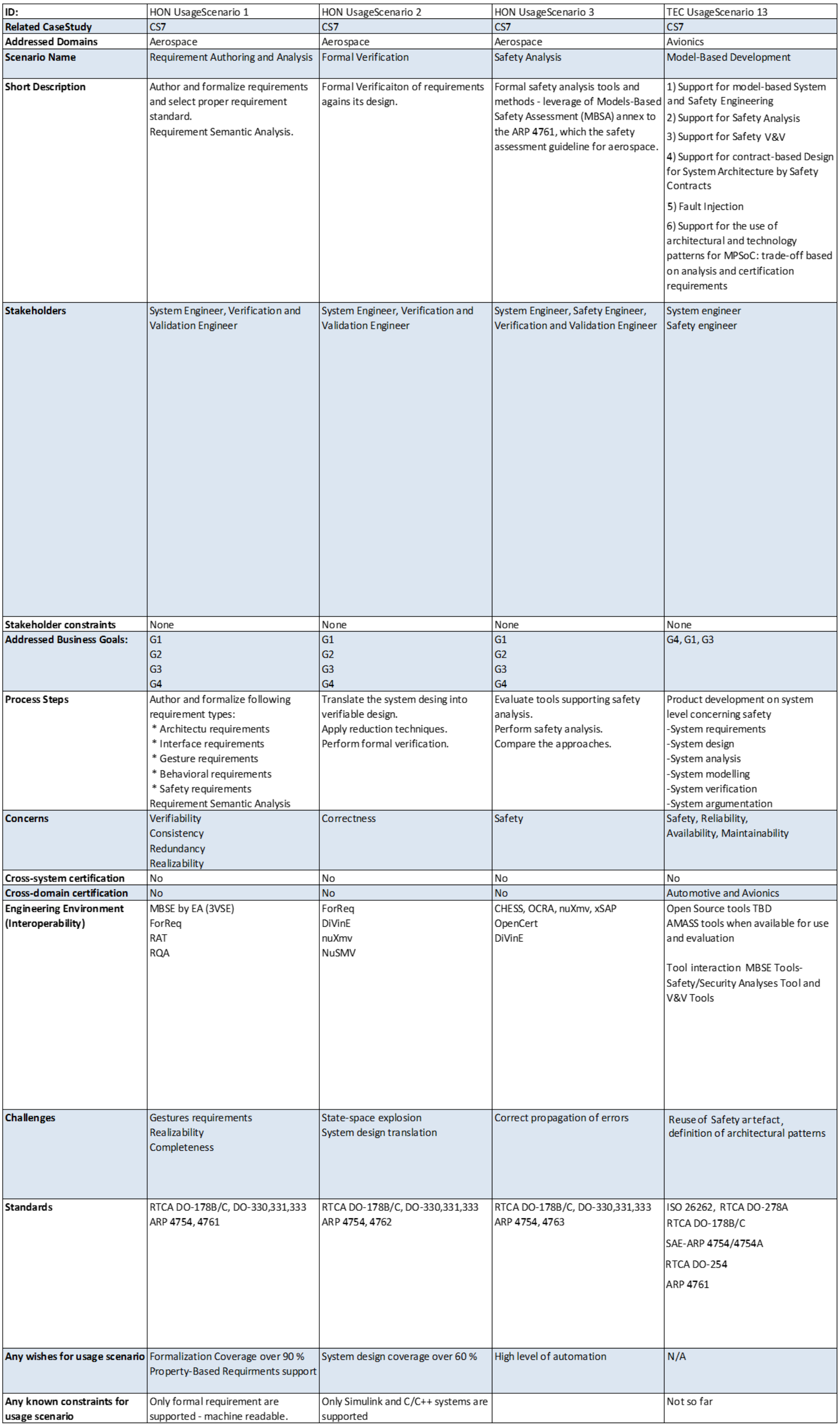
> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

- Methodology and tools to semantically analyse the requirements.
- Methodology and tools to formally verify the requirements against system architecture and system design.
- Methodology and tools to perform safety assessment.

2.7.7 Usage scenarios

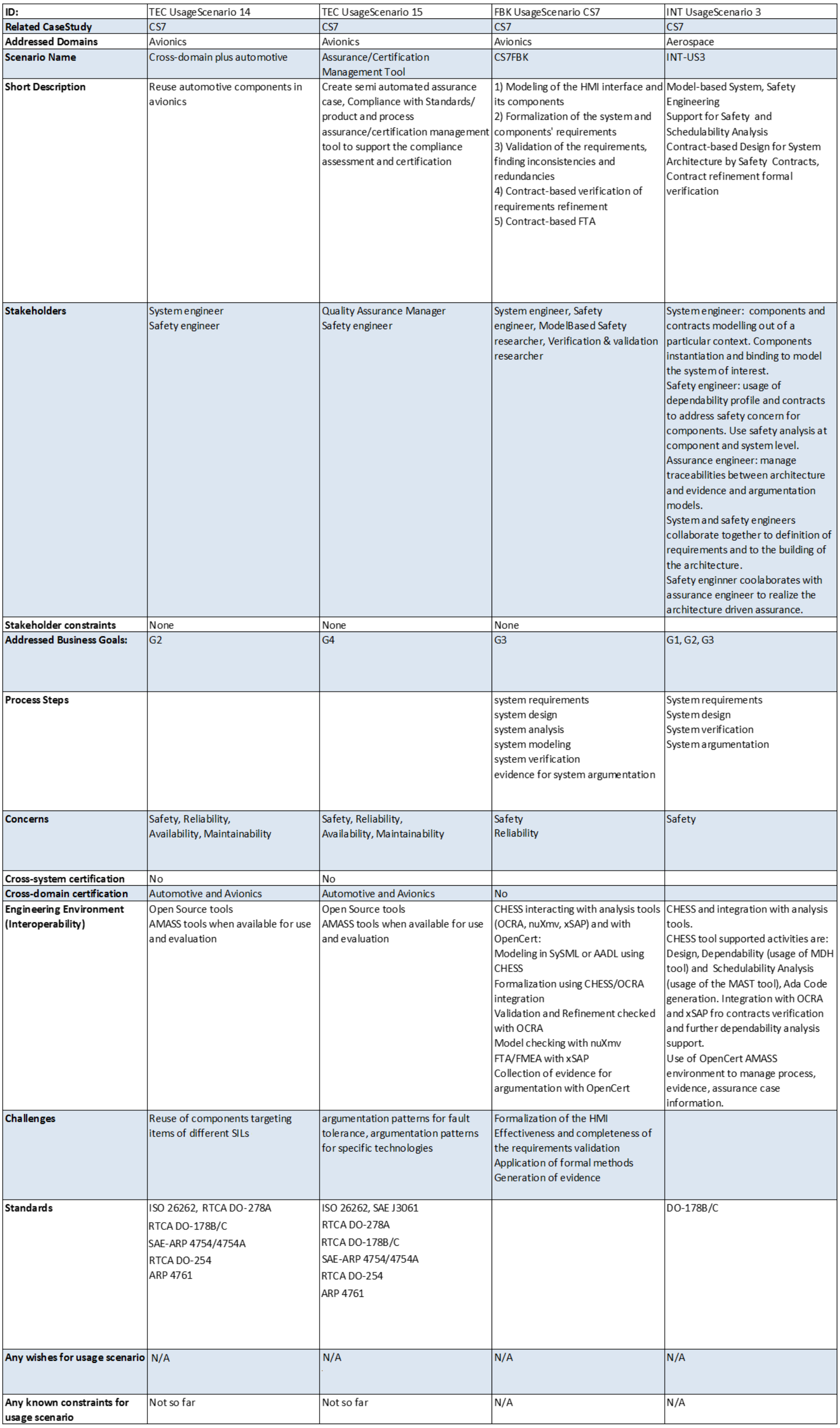
Table 24, Table 25 and Table 26 show the 11 usage scenarios related to Case Study 7.

Table 24. CS7 HON and TEC usage scenarios



ID:	HON UsageScenario 1	HON UsageScenario 2	HON UsageScenario 3	TEC UsageScenario 13
Related CaseStudy	CS7	CS7	CS7	CS7
Addressed Domains	Aerospace	Aerospace	Aerospace	Avionics
Scenario Name	Requirement Authoring and Analysis	Formal Verification	Safety Analysis	Model-Based Development
Short Description	Author and formalize requirements and select proper requirement standard. Requirement Semantic Analysis.	Formal Verificaiton of requirements agains its design.	Formal safety analysis tools and methods - leverage of Models-Based Safety Assessment (MBSA) annex to the ARP 4761, which the safety assessment guideline for aerospace.	1) Support for model-based System and Safety Engineering 2) Support for Safety Analysis 3) Support for Safety V&V 4) Support for contract-based Design for System Architecture by Safety Contracts 5) Fault Injection 6) Support for the use of architectural and technology patterns for MPSoC: trade-off based on analysis and certification requirements
Stakeholders	System Engineer, Verification and Validation Engineer	System Engineer, Verification and Validation Engineer	System Engineer, Safety Engineer, Verification and Validation Engineer	System engineer Safety engineer
Stakeholder constraints	None	None	None	None
Addressed Business Goals:	G1 G2 G3 G4	G1 G2 G3 G4	G1 G2 G3 G4	G4, G1, G3
Process Steps	Author and formalize following requirement types: * Architectu requirements * Interface requirements * Gesture requirements * Behavioral requirements * Safety requirements Requirement Semantic Analysis	Translate the system desing into verifiable design. Apply reduction techniques. Perform formal verification.	Evaluate tools supporting safety analysis. Perform safety analysis. Compare the approaches.	Product development on system level concerning safety -System requirements -System design -System analysis -System modelling -System verification -System argumentation
Concerns	Verifiability Consistency Redundancy Realizability	Correctness	Safety	Safety, Reliability, Availability, Maintainability
Cross-system certification	No	No	No	No
Cross-domain certification	No	No	No	Automotive and Avionics
Engineering Environment (Interoperability)	MBSE by EA (3VSE) ForReq RAT RQA	ForReq DiVinE nuXmv NuSMV	CHESS, OCRA, nuXmv, xSAP OpenCert DiVinE	Open Source tools TBD AMASS tools when available for use and evaluation Tool interaction MBSE Tools-Safety/Security Analyses Tool and V&V Tools
Challenges	Gestures requirements Realizability Completeness	State-space explosion System design translation	Correct propagation of errors	Reuse of Safety artefact, definition of architectural patterns
Standards	RTCA DO-178B/C, DO-330,331,333 ARP 4754, 4761	RTCA DO-178B/C, DO-330,331,333 ARP 4754, 4762	RTCA DO-178B/C, DO-330,331,333 ARP 4754, 4763	ISO 26262, RTCA DO-278A RTCA DO-178B/C SAE-ARP 4754/4754A RTCA DO-254 ARP 4761
Any wishes for usage scenario	Formalization Coverage over 90 % Property-Based Requirments support	System design coverage over 60 %	High level of automation	N/A
Any known constraints for usage scenario	Only formal requirement are supported - machine readable.	Only Simulink and C/C++ systems are supported		Not so far

Table 25. CS7 TEC, FBK and INT usage scenarios



ID:	TEC UsageScenario 14	TEC UsageScenario 15	FBK UsageScenario CS7	INT UsageScenario 3
Related CaseStudy	CS7	CS7	CS7	CS7
Addressed Domains	Avionics	Avionics	Avionics	Aerospace
Scenario Name	Cross-domain plus automotive	Assurance/Certification Management Tool	CS7FBK	INT-US3
Short Description	Reuse automotive components in avionics	Create semi automated assurance case, Compliance with Standards/ product and process assurance/certification management tool to support the compliance assessment and certification	1) Modeling of the HMI interface and its components 2) Formalization of the system and components' requirements 3) Validation of the requirements, finding inconsistencies and redundancies 4) Contract-based verification of requirements refinement 5) Contract-based FTA	Model-based System, Safety Engineering Support for Safety and Schedulability Analysis Contract-based Design for System Architecture by Safety Contracts, Contract refinement formal verification
Stakeholders	System engineer Safety engineer	Quality Assurance Manager Safety engineer	System engineer, Safety engineer, ModelBased Safety researcher, Verification & validation researcher	System engineer: components and contracts modelling out of a particular context. Components instantiation and binding to model the system of interest. Safety engineer: usage of dependability profile and contracts to address safety concern for components. Use safety analysis at component and system level. Assurance engineer: manage traceabilities between architecture and evidence and argumentation models. System and safety engineers collaborate together to definition of requirements and to the building of the architecture. Safety enginner coolaborates with assurance engineer to realize the architecture driven assurance.
Stakeholder constraints	None	None	None	
Addressed Business Goals:	G2	G4	G3	G1, G2, G3
Process Steps			system requirements system design system analysis system modeling system verification evidence for system argumentation	System requirements System design System verification System argumentation
Concerns	Safety, Reliability, Availability, Maintainability	Safety, Reliability, Availability, Maintainability	Safety Reliability	Safety
Cross-system certification	No	No		
Cross-domain certification	Automotive and Avionics	Automotive and Avionics	No	
Engineering Environment (Interoperability)	Open Source tools AMASS tools when available for use and evaluation	Open Source tools AMASS tools when available for use and evaluation	CHESS interacting with analysis tools (OCRA, nuXmv, xSAP) and with OpenCert: Modeling in SySML or AADL using CHESS Formalization using CHESS/OCRA integration Validation and Refinement checked with OCRA Model checking with nuXmv FTA/FMEA with xSAP Collection of evidence for argumentation with OpenCert	CHESS and integration with analysis tools. CHESS tool supported activities are: Design, Dependability (usage of MDH tool) and Schedulability Analysis (usage of the MAST tool), Ada Code generation. Integration with OCRA and xSAP fro contracts verification and further dependability analysis support. Use of OpenCert AMASS environment to manage process, evidence, assurance case information.
Challenges	Reuse of components targeting items of different SILs	argumentation patterns for fault tolerance, argumentation patterns for specific technologies	Formalization of the HMI Effectiveness and completeness of the requirements validation Application of formal methods Generation of evidence	
Standards	ISO 26262, RTCA DO-278A RTCA DO-178B/C SAE-ARP 4754/4754A RTCA DO-254 ARP 4761	ISO 26262, SAE J3061 RTCA DO-278A RTCA DO-178B/C SAE-ARP 4754/4754A RTCA DO-254 ARP 4761		DO-178B/C
Any wishes for usage scenario	N/A	N/A	N/A	N/A
Any known constraints for usage scenario	Not so far	Not so far	N/A	N/A

Table 26. CS7 UOM and LAN usage scenarios

ID:	UOM UsageScenario 1	LAN UsageScenario 1	LAN UsageScenario 2
Related CaseStudy	CS7	CS7	CS7
Addressed Domains	Aerospace	Aviation	Aviation
Scenario Name	Formal Verification	Reuse of automotive components	Tool chain
Short Description	Formal Verificaiton of requirements agains its design. Note that we are not an industrial partner, however our tools are used by HON (industrial AMASS partner). This usage scenario is taken from HON UsageScenario 2.	Reuse of safety artefacts of automotive components in the aviation domain.	Investigation how AMASS tool platform supports the usage of automotive components in terms of evidence management, traceability and seamless integration
Stakeholders	System Engineer, Verification and Validation Engineer	Avionics Engineer Safety Engineer	System Engineer Avionics Engineer Verification Engineer
Stakeholder constraints	None	None	None
Addressed Business Goals:	G1 G2 G3 G4	G2	G4
Process Steps	Translate the system desing into verifiable design. Apply reduction techniques. Perform formal verification.	> Identifying safety artefacts of automotive components > check mapping for the aviation domain > use AMASS tools for evident management	Product development on System / Component level - Requirements Engineering - Design - Engineering / Coding / Testing - Verification
Concerns	Correctness	Safety, Reliability	Safety, Reliability
Cross-system certification	No	N/A	N/A
Cross-domain certification	No	Automotive and Avionics	Automotive and Avionics
Engineering Environment (Interoperability)	DiVinE	AMASS Tools as soon as available, Open Source Tools TBD	AMASS Tools as soon as available, Open Source Tools TBD
Challenges	State-space explosion System design translation	N/A	N/A
Standards	RTCA DO-178B/C, DO-330,331,333 ARP 4754, 4762	RTCA DO-178C, DO-254, DO-160, ARP4761, ARP4754	RTCA DO-178C, DO-254, DO-160, ARP4761, ARP4754
Any wishes for usage scenario	System design coverage over 60 %	no	no
Any known constraints for usage scenario	Only Simulink and C/C++ systems are supported	no	no

2.8 CS8: Telematics Function

2.8.1 Short description of the case study

This case study will investigate a telematics function providing *position and time*, and which is aimed at use in automated and connected vehicles. For automated vehicles, positioning may be used as part of the autonomous drive (AD) pilot function, which has implications for safety. However, functional safety is also increasingly susceptible to security threats from malicious adversaries. There are multiple attack vectors, including direct tampering with the equipment and remote attacks. Wireless communication and positioning using global navigation satellite systems (GNSS) are examples where an attacker can pose a threat even without physical access to the vehicle. Positioning can be provided using GNSS and, in order to increase precision, additional information provided using vehicle-to-vehicle or vehicle-to-infrastructure communication (V2X) or cloud-based services.

Current prototype AD vehicles typically rely on a range of sensors, to provide both positioning and object detection. These sensors are often expensive, but for mass-market introduction, the cost of components will be an important factor. Thus, there will be a demand for reducing the amount and cost of sensors. Positioning that relies on GNSS and V2X may be a cost-effective component to provide position and accurate time for the AD pilot, if the safety requirements can be met. The case study will focus on remote attacks and the security implications for safety, as well as the trade-offs between safety, security and performance for the positioning function.

2.8.2 Technical description of the case study

Figure 24 show a logical view of a telematics function with tree different wireless interfaces. IF-1 is a GNSS signal. IF-2 and IF-3 may be data from other vehicles or cloud-based services, for instance via cellular networks such as 4G/LTE or vehicular ad hoc networks (VANETs) such as 802.11p, using communication standards such as ETSI C-ITS or OEM proprietary interfaces. Within the vehicle, a consumer function (C) receives information from the wireless network interface provider (P). The consumer can act on the physical environment by releasing or withholding energy, e.g. accelerating, braking or steering a vehicle. If these actions are based on the information provided by P, they can be dangerous for the environment i.e. safety critical. If so, this system should be subject to a functional safety analysis.

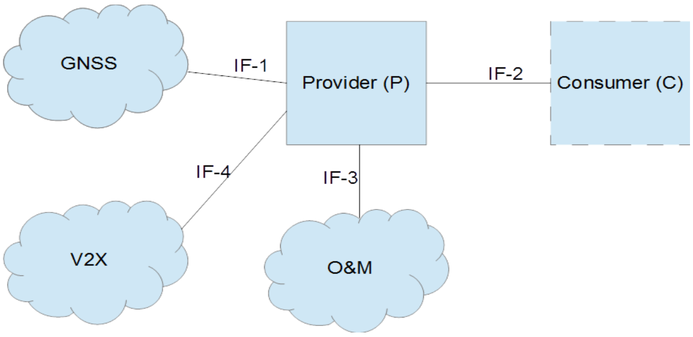


Figure 24. Logical view of telematics functions for a connected vehicle.

The case study will consider safety and security concerns focused on the wireless interfaces, which can be the subject of both malicious attacks and other failures. The aim of the demonstrator is to act as a testbed for a multi-concern assurance case and investigations on verification of safety and security related mechanisms. Therefore, it will be gradually evolved in three iterations based on the needs defined in the previous iteration.

2.8.2.1 First iteration (Core)

For the first (core) iteration of AMASS a very simple telematics function, on which to base a first multi-concern assurance case, will be used. The function is described in Figure 25. It consists of a GPS receiver and (simulated) wheel speed sensor which give input to a function block (implemented in a Raspberry Pi single-board computer) acting as producer (P) of *position*, *time*, and *ok/nok* signals to consumer functions. The ok/nok signal will show if the position and time signals are currently reliable. An example of a malicious attack can be jamming or spoofing of a GPS signal, which at a typical signal power of −127.5 dBm is highly susceptible to such attacks.

The function will be developed and assessed as a safety element out of context (SEooC) with a safety and a cybersecurity case.

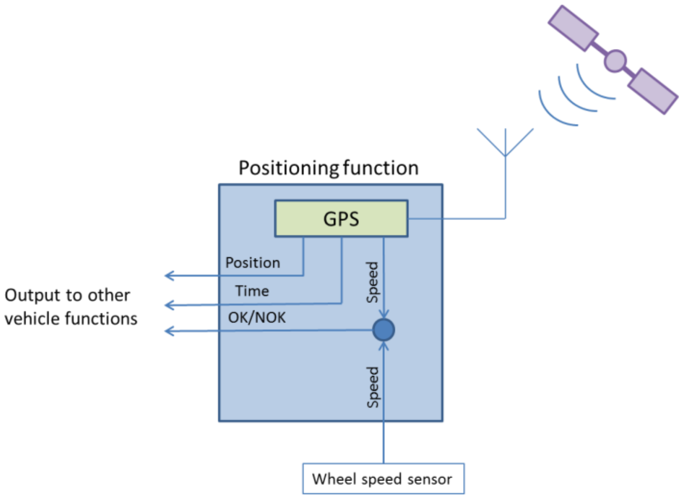


Figure 25. Positioning function for Core iteration.

2.8.2.2 Second and third iteration (P1 and P2)

For the second and third iterations the case study will be extended. The goal for the third iteration is to provide an industry-relevant function based on current standards. The specifications will be refined based on the findings in the previous iterations but the current plan, illustrated in Figure 26, includes:

- RTK-GPS for accurate positioning. This involves an additional wireless interface (e.g. 4G/LTE cellular) to provide internet access to RTK reference stations. This additional interface presents another possible attack vector for a malicious adversary. Also, IMU for improved correlation possibilities in the positioning function.
- Integrating an ITS facility that sends CAM messages according to the ETSI C-ITS standard. Adding a second element will also include ensuring the compatibility between the two SEooC components: the positioning function and the ITS facility. The aim is to use AMASS component contracts (to be developed in WP3 and WP4) for this purpose.

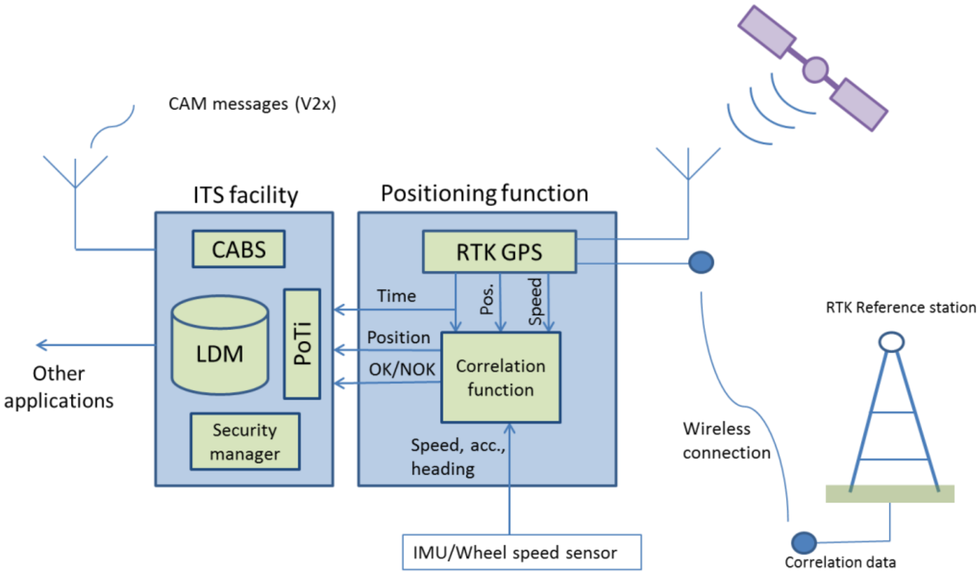


Figure 26. Positioning function and ITS facility for P1/P2 iterations.

2.8.3 Case study state of the art

For functional safety in the automotive domain the ISO 26262:2011 standard is used by all major OEMs. For security, the SAE J3061 *Cybersecurity Guidebook for Cyber-Physical Vehicle Systems* was recently released (January 2016) and, while not an international standard, it is the so far most ambitious work to create a systematic security process. The framework is also similar to that of ISO 26262. An ISO working group has been initiated to create an international standard for security for road vehicles and may base this work on J3061. In addition, many research projects and methods from the research community have been working on methods for managing the security in the automotive domain, for instance the EVITA and HEAVENS projects and the SAHARA methodology.

Furthermore, the standards for differential GNSS (RTK-GPS), and the standard for C-ITS may be used in the case study.

2.8.4 Case study state of the practice

The positioning function was originally developed in an iterative fashion for use in non-safety-critical self-driving test vehicles, i.e. without safety practices. An improved version is under development following most (but not all) of ISO 26262 and SAE J3061 with traditional development methods as described in this section. This version will constitute the baseline for the case study.

2.8.4.1 Workflow

The workflow is illustrated in Figure 27, and logically follows the V-models for both safety (according to ISO 26262) and security (according to SAE J3061). In practice, the work is performed iteratively and not in a waterfall fashion. Hardware (i.e. 26262, part 5) is currently omitted in the safety work. The safety and security cases are created and managed independently.

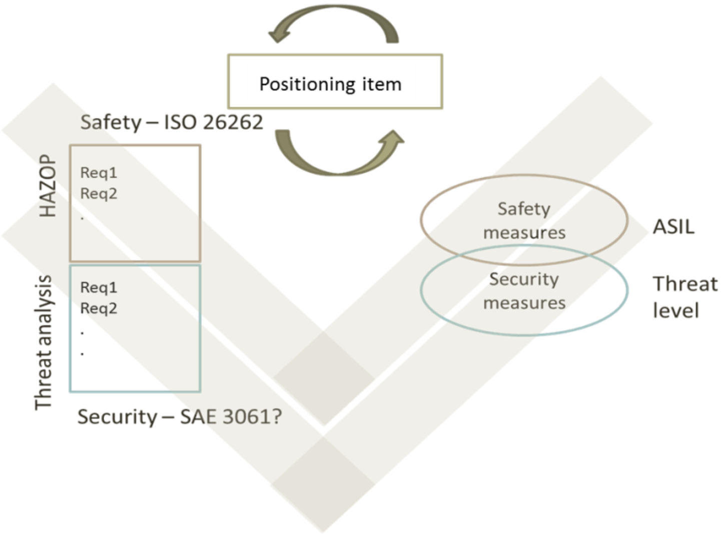


Figure 27. V-model for safety and security

2.8.4.2 Assessment

The goal for this system is not certification, as it is used for test purposes and is not a commercial product. The assurance activities are performed to increase quality of the system, i.e. the measures prescribed by the ISO 26262 and J3061 standards are applied. However, the safety and security cases are not complete; for instance, hardware and tool qualification is not included. Partial assessment is carried out internally by persons within the developing organizations that have not been directly involved with development of the system, but have assessment competence. This means there is some degree of independence, though not to the extent needed for actual certification.

2.8.4.3 Involved roles

The following roles are involved in:

- Development team – Develops the system and is responsible for architecture, development, test and documentation.
- Functional safety manager – Responsible for functional safety assurance activities.
- Cybersecurity manager – Responsible for security assurance activities.
- Assessors.

The functional safety and security managers are also part of the development team. The assessment is partial and the objective for assessment is explained in Section 2.8.4.2.

2.8.4.4 Tools and tool chains

2.8.4.4.1 Used tools and methods (included guidelines)

For development, mainly open source tools are used. No qualified tools or dedicated assurance tools are currently used. Use of better methods and tools is an expected result from the AMASS project.

2.8.4.4.2 Tool chain

The following tools are used:

- *Requirements:* MS Excel
- *System documentation:* MS Word, MS Excel
- *Configuration and change management:* SVN, Git, Jira
- *SW and test documentation:* Doxygen
- *SW development:* Eclipse, CMake, Jenkins

- *SW test:* CTest, CUnit, PC-Lint

2.8.5 Expected technical improvements

2.8.5.1 STO1. Architecture-driven Assurance

- N/A

2.8.5.2 STO2. Multi-concern Assurance

- Improved methods to analyse, create assurance cases, and assess systems with multiple concerns, specifically for interplay between safety, security, and performance.

2.8.5.3 STO3. Seamless Interoperability

- Provide feedback to tool providers on tool efficiency and tool interoperability for use in a multi-concern assurance case.

2.8.5.4 STO4. Cross/Intra-Domain Reuse

- Re-use of e.g. analysis and verification results between concerns in a multi-concern assurance case.

2.8.6 Business needs

2.8.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

Methodology and tools for efficient management of multi-concern assurance, e.g. ISO 26262, SAE J3061 and possibly additional standards. Multi-concern verification including re-use for efficiency.

2.8.6.2 AMASS Goal 2

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*

Methodology for handling interplay between concerns and/or re-use between concerns for multi-concern assurance and assessment for multiple standards.

2.8.6.3 AMASS Goal 3

> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

N/A

2.8.6.4 AMASS Goal 4

> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

N/A

2.8.7 Usage scenarios

Table 27 shows the 3 usage scenarios related to Case Study 8.

Table 27. CS8 SPS and COM usage scenarios

ID:	SPS UsageScenario 1	SPS UsageScenario 2	COM UsageScenario 1
Related CaseStudy	CS8	CS8	CS8
Addressed Domains	Automotive	Automotive	Automotive
Scenario Name	MCAC	MCASS	SAASSA
Short Description	Creating multi-concern assurance case focusing on safety and security, in particular security impact on safety. Use of existing standards ISO 26262 and SAE J 3061 (HEAVENS security model). Suitability of standards and tools, multi-concern	Multi-concern assessment. Practicability and efficiency of co-assessment of several standards for the same product based on a multi-concern assurance case.	Multi concern Specification, Analysis and Assurance of Safety, Security and Availability for a CPS-subsystem . The practicality and suitability of existing standards and development tools will be investigated when
Stakeholders	System engineer Functional safety manager Security manager? Safery engineer Security engineer Test manager	Assessors (safery assessor and security assessor) Functional safety manager Test manager Security manager?	System Engineer Safety Engineer Security Engineer Verification&Validation Engineer Safety Manager Securiy Manager Safety Assessor Security Assessor
Stakeholder constraints	None	None	None
Addressed Business Goals:	G1, (G2)	G1	G1
Process Steps	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification -System argumentation	Assessment	Product development on system level concerning safety/security/availability -System requirements -System design -System analysis -System modelling -System verification
Concerns	Safety and security, possibly also influence on safety of other concerns such as availability and	Safety and Security	Safety and Security and Function Availability
Cross-system certification	No	No	No
Cross-domain certification	No	No	No
Engineering Environment (Interoperability)	Open source tools, OpenCert and use/evaluation of tools from other AMASS partners. Modifi (internal	TBD	Open Source tools TBD AMASS tools when available for use and evaluation
Challenges	Safety and Security co engineering	Safety and Security co assessment	Safety and Security and Function Availability co-engineering
Standards	ISO 26262 SAE J 3061 IEC 62443 (possibly?)	ISO 26262 SAE J 3061 IEC 62443 (possibly?)	ISO 26262 SAE J3061
Any wishes for usage scenario	Use of AMASS partner tools for assurance case.	No	No
Any known constraints for usage	No	No	No

2.9 CS9: Safety-Critical SW Lifecycle of a Monitoring System for NavAid (ATM domain)

2.9.1 Short description of the case study

Within Air Traffic Management (ATM) domain, the radio-navigation equipment (often defined NavAids and including radio-beacons such as DME, TACAN, VOR, ILS etc.) are currently the most widespread systems for providing aircrafts with exact location in space and time. They are CPS based on the joint contribution from the physical electromagnetic fields which govern the positioning mechanism and sophisticated computation processes.

Among such systems, the DME system is a Distance Measuring Equipment which provides pilots with distance information between the aircraft and the location of the DME ground equipment. Basically, the airborne DME transmitter interrogates the DME ground station, which replies after a fixed and known delay. An additional, variable delay is proportional to the distance between the airborne interrogator and the ground station: from this variable delay it is possible to compute such distance. The system is used for both en-route and terminal area guidance.

DME, as well as other navaid systems, is subject to the strict ICAO (International Civil Aviation Organization) accuracy requirements and to severe constraints in terms of service integrity/continuity/availability. This makes some aspects of DME design technology (requirement-to-design mapping, testing, validation, certification) predominate issues. This is especially true for the core subsystem dedicated to assure the integrity of the system, the Monitoring subsystem: it measures the quality and the performance of the radiated signal, as well as the internal parameters of the equipment. On the basis of such assessment the subsystem automatically and autonomously defines the reliability of the positioning service provided to aircrafts, extending such assessment to making the service unavailable.

For a safety-critical system such as DME, model-based formal approaches to the validation and verification of SW design (and re-design) represent an answer to the issues mentioned above and result in an increase in overall safety and maintainability of such CPS.

The ATM dept. of Thales Italia (THI) will drive an industrial case study aimed to re-engineer, through the usage of tools and methods provided by the AMASS project, both the SW of the DME Monitoring subsystem and the SW development processes, applying the CNS/ATM safety certification standards (EUROCAE ED-109', 'RTCA DO-278', 'EUROCAE ED-153').

2.9.2 Technical description of the case study

The new approach, provided by AMASS for the THI CS, to the lifecycle management process will be evaluated applying specific process metrics and compared (in terms of effort, quality and safety assurance) to the previous generation approach: improvement will be estimated by analysing current and historical process effort records. This will be also beneficial in assessing the maturity of the framework methodologies and of the relevant tools (e.g.: automatic “safe” code generation, automatic tests generation etc.). The focus with associated contributors (such as INT, TEC, etc.) will thus be on requirement analysis, design (modelling, simulation), early validation, and performance verification phases. The SW development process, in order to incorporate the new methodologies provided by AMASS framework, will be also focused, in association with the contributors, on the modular and on the incremental certification on the product subsystems when safety certification standards apply (EUROCAE ED-109', 'RTCA DO-278', 'EUROCAE ED-153').

The methods and the tools, associated to the main phases of the processes just mentioned, can be summarized as follows:

1. Requirement analysis phase: supported by formal methods for requirements definition and by tools for requirement analysis. Resulting artefact: SRS (Software Requirements Specification).

2. Software design phase: supported by tools for software modelling and software simulation. Formal methods for early model validation and advanced tools for requirements-to-design traceability will also contribute to this phase.
3. Software implementation phase: supported by model-to-code translation tool for automatic code generation.
4. Software verification phase: supported by tools for automatic test case generation (including component testing, integration testing and formal tests) and by tools for code-coverage testing. The collection of result artefacts will result in a Test Report.

THI CS will benchmark:

a. Tools and methodologies for satisfying both the safety requirements and the performance of the related processes.
b. Methodology for developing software requirements and a resulting architecture that reduces certification effort (and re-certification effort, in case of code changes).
c. Tools and metrics for performance analysis.
d. Tools and methodologies for demonstrating and verifying the compliance of the architecture with the requirements and of the implementation with the architecture (required for safety aspects).
e. Tools and methodologies for demonstrating and verifying the traceability of test cases with source code and requirements (required for safety aspects).
f. Early verification methods to anticipate the detection of possible problems especially linked to safety and performance requirements.

Moving to model-based development (reference to DO-278A model-based design supplement) and to O.O./C++ (reference to DO-278A O.O. supplement) and the usage of Automatic “safe” Code and Tests Generation Tool, is expected to help in achieving the aforesaid goals.

2.9.3 Case study state of the art

According to the EUROCONTROL Safety Assessment Methodology (SAM), the complete software-lifecycle safety assurance is covered by the following ATM regulations, norms and standards:

- RTCA Inc. DO-178B. Software Considerations in Airborne Systems and Equipment Certification. RTCA Inc. / EUROCAE. DO-178B/ED-12B. 1992.
- RTCA, EUROCAE. DO-278 / ED-109. Guidelines for Communication, Navigation, Surveillance, and Air Traffic Management (CNS/ATM) Systems Software Integrity Assurance. RTCA Inc. / EUROCAE. DO-278/ED-109. 3/5/2002.
- RTCA Inc. / EUROCAE. DO-278A/ED-109A. Software Integrity Assurance Considerations for Communication, Navigation, Surveillance and Air Traffic Management (CNS/ATM) Systems. December 2011.
- Eurocontrol. ESARR6. Eurocontrol Safety Regulatory Requirement 6 Software in ATM Functional Systems. May 2010.
- Eurocontrol. ESARR4. Eurocontrol Safety Regulatory Requirement 4 Risk Assessment and Mitigation in ATM. April 2001.
- EUROCAE. ED-153. Guidelines for ANS Software Safety Assurance. August 2009.
- RTCA Inc. / EUROCAE. DO-178C / ED-12B. Software Considerations in Airborne Systems and Equipment Certification. December 2011.
- RTCA Inc. / EUROCAE. DO-330 / ED-215. Software Tool Qualification Considerations. December 2011 - January 2012.
- RTCA Inc. / EUROCAE. DO-331 / ED-216. Model-Based Development and Verification Supplement to DO-178C and DO-278A / Model-Based Development and Verification Supplement to ED-12B and ED-109A. December 2011 - January 2012.

- RTCA Inc. / EUROCAE. DO-332 / ED-217. Object-Oriented Technology and Related Techniques Supplement to DO-178C and DO-278A / Object-Oriented Technology and Related Techniques Supplement to ED-12C and ED-109A. December 2011 - January 2012.
- RTCA Inc. DO-333 / ED-218. Formal Methods Supplement to DO-178C and DO-278A / Model-Based Development and Verification Supplement to ED-12C and ED-109A. December 2011.

2.9.4 Case study state of the practice

The software development lifecycle is in accordance with the DO-254 / ED-109 objectives:

- Software plans and standards are defined.
- System requirements, through system functional architecture, are allocated to hardware and software items.
- Software requirements cover all the system requirements and the functions defined in system architecture.
- Software architecture is defined according to the software requirements and then, depending on assurance level, low level software requirements are defined.
- Test procedures are defined to cover all the high and low level requirements: the code coverage is performed only when required by the assurance level.
- External and internal ICDs are defined for each software item.
- Reviews with all the involved stakeholders are performed for all the documents and for the source code (when needed).

The current lifecycle, described above, shall benefit from the tools provided by AMASS and currently not (or partially) available, according to the workflow represented in Figure 28.

2.9.4.1 Workflow

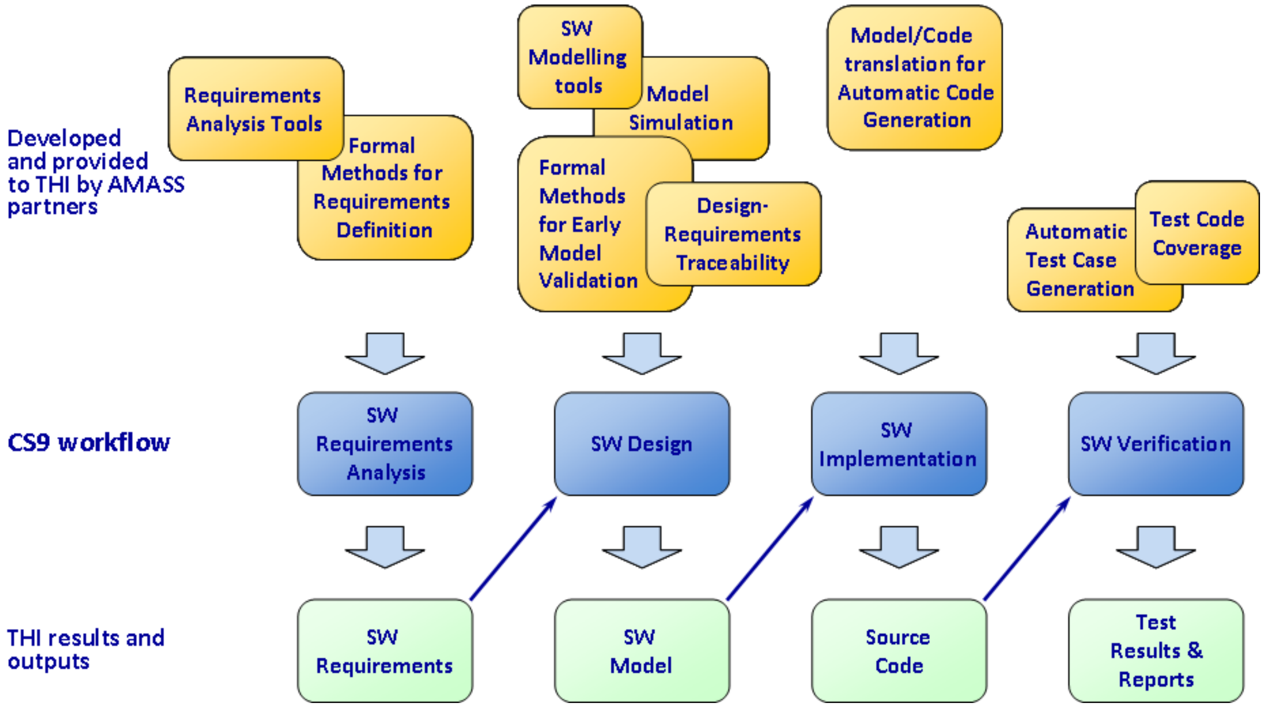

Developed and provided to THI by AMASS partners
Requirements Analysis Tools
Formal Methods for Requirements Definition
SW Modelling tools
Model Simulation
Formal Methods for Early Model Validation
Design-Requirements Traceability
Model/Code translation for Automatic Code Generation
Automatic Test Case Generation
Test Code Coverage
CS9 workflow
SW Requirements Analysis
SW Design
SW Implementation
SW Verification
THI results and outputs
SW Requirements
SW Model
Source Code
Test Results & Reports

Figure 28. CS9 workflow

2.9.4.2 Assessment

Not applicable.

2.9.4.3 Involved roles

The following job roles will be involved in the CS9:

- System Engineers and SW Engineers in the Requirement Analysis phase
- SW Engineers and Safety Engineers in the Design phase
- SW Engineers and HW Engineers in the Implementation phase
- System Engineers, SW Engineers and HW Engineers in the Verification phase

2.9.4.4 Tools and Tool chains

2.9.4.4.1 Used tools and methods (included guidelines)

- *DOORS* is used for the system requirements, software requirements, architecture and test procedures.
- *Rhapsody* is used for system and software architecture and partially for automatic code generation.
- *IVV Manager* tool for test campaign management will be used in the next months.
- Software configuration management tool in use is *ClearCase* in conjunction with *Bugzilla* for problem reporting.

2.9.4.4.2 Tool Chain

The tool chain has been already represented in Figure 28, where methods and the tools are associated to the main phases of the software development process:

- Design-Requirements Traceability is currently managed, inside THI, through *DOORS.*
- SW Modelling and Model/Code translation for Automatic Code Generation are currently managed, inside THI, through *Rhapsody*; already available (inside AMASS) alternatives could be *CHESS* plus *OCRA* (which include formal methods for defining contracts among architectural components).
- *CHESS* plus *OCRA* should be able to perform also a Fault Tree Analysis during the early phases of the workflow represented in Figure 28.

All the (other) tools/methods are expected to be provided by AMASS Project.

The information exchange between *DOORS* and *RHAPSODY* takes place through text documents.

ClearCase is just a repository for the various software releases.

IVV Manager stores text reports too.

None of the mentioned tools is qualified.

2.9.5 Expected technical improvements

Improvements are expected in terms of:

- reduced design effort (cost)
- time-to-market
- system performance

The technical objects, assessed within the case study will be:

- STO1 (System Architecture-driven Assurance)
- STO2 (Multi-concern assurance)

2.9.5.1 STO1. Architecture-driven Assurance

- Architecture-driven modelling to fulfil both technical/functional and dependability requirements.
- Assurance patterns specification for compliance with standards and early introduction of safety concerns.

2.9.5.2 STO2. Multi-concern Assurance

- Safety assurance automation through requirement analysis tools.
- Improvement of V&V phase for safety requirements.
- Complete traceability from the DO-278 / ED-109 objectives to the source code through all the artefacts.

2.9.5.3 STO3. Seamless Interoperability

- N/A

2.9.5.4 STO4. Cross/Intra-Domain Reuse

- N/A

2.9.6 Business needs

2.9.6.1 AMASS Goal 1

G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.

- Tools and methods for the early introduction, into the development process, of safety requirements.
- Methods for early model validation and for verification; tools for code-coverage testing.

2.9.6.2 AMASS Goal 2

G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.

- Methods, for generating software architecture and requirements, which can reduce certification effort.
- Tools for automatic generation of reports, checklists and evidences to support the certification. Automatic check to verify that all the DO-278 / ED-109 objectives have been satisfied.

2.9.6.3 AMASS Goal 3

G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.

- N/A

2.9.6.4 AMASS Goal 4

G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.

- N/A

2.9.7 Usage scenarios

Table 28 and Table 29 show the 5 usage scenarios related to Case Study 9.

Table 28. CS9 THI usage scenarios

ID:	THI UsageScenario 1	THI UsageScenario 2	THI UsageScenario 3
Related CaseStudy	CS9	CS9	CS9
Addressed Domains	ATM (Air Traffic Management)	ATM (Air Traffic Management)	ATM (Air Traffic Management)
Scenario Name	SWRA	SWD	SWI
Short Description	From System to SW requirements by using the following methods and tools provided by AMASS: • formal methods for requirements definition • requirements analysis tools	From SW requirements to SW design by using the following methods, tools and models provided by AMASS: • SW modelling tools • models simulation • tools for requirements traceability into design • formal methods for early	SW implementation supported by a model-to-code translation tool provided by AMASS, for automatic code generation.
Stakeholders	N/A	N/A	N/A
Stakeholder constraints	N/A	N/A	N/A
Addressed Business Goals:	G1 G3	G1 G3	G1 G3
Process Steps	System requirements. SW requirements.	SW modelling. SW design.	SW implementation.
Concerns	Safety.	Safety.	Safety.
Cross-system certification	-	-	-
Cross-domain certification	No	No	No
Engineering Environment (Interoperability)	Doors	- Microsoft Visio - Rhapsody	- Eclipse - Thales Control - ClearCase - Bugzilla
Challenges	Improved safety assurance.	Improved SW-HW co-design.	Automation of code generation.
Standards	EUROCAE ED-109', 'RTCA DO-278', 'EUROCAE ED-153'	EUROCAE ED-109', 'RTCA DO-278', 'EUROCAE ED-153'	EUROCAE ED-109', 'RTCA DO-278', 'EUROCAE ED-153'
Any wishes for usage scenario	Use of AMASS-partners tools and methods to improve safety assurance and development efficiency.	Use of AMASS-partners tools and methods to improve safety assurance and development efficiency.	Use of AMASS-partners tools and methods to improve safety assurance and development efficiency.
Any known constraints for	N/A	N/A	C/C++ only.

Table 29. CS9 THI and INT usage scenarios

ID:	THI UsageScenario 4	INT UsageScenario 2
Related CaseStudy	CS9	CS9
Addressed Domains	ATM (Air Traffic Management)	ATM
Scenario Name	SWV	INT-US2
Short Description	SW verification and validation supported by the following tools and methods provided by AMASS: • automatic test code generator • test code coverage assessment • complete traceability from	Model-based System, Safety Engineering Support for Safety and Schedulability Analysis Contract-based Design for System Architecture by Safety Contracts, Contract refinement formal verification
Stakeholders	N/A	System engineer Safety engineer System and safety engineers collaborate together to definition of requirements and to the building of the architecture. Safety enginner coolaborates with assurance engineer to
Stakeholder constraints	N/A	N/A
Addressed Business Goals:	G1 G3	G1, G2, G3
Process Steps	SW verification. SW certification.	System requirements System design System verification System argumentation
Concerns	Safety.	Safety
Cross-system certification	-	N/A
Cross-domain certification	No	
Engineering Environment (Interoperability)	IVV Manager	CHESS and integration with analysis tools. CHESS tool supported activities are: Design, Dependability (usage of MDH tool) and Schedulability Analysis (usage of the MAST
Challenges	Automation of verification and validation process.	N/A
Standards	EUROCAE ED-109', 'RTCA DO-278', 'EUROCAE ED-153'	DO-178B/C
Any wishes for usage scenario	Use of AMASS-partners tools and methods to improve safety assurance and development efficiency.	N/A
Any known constraints for	N/A	N/A

2.10 CS10: Certification basis to boost the usage of MPSoC architectures

2.10.1 Short description of the case study

The target of this Case Study is to prove the validity of the different architectures and related development methodologies and tool chains proposed by AMASS project and previous projects such as OPENCOSS and SafeCer, opening new application domains to the use of multicores. This use case will be clearly targeted to a final product application and, therefore, it must be guaranteed not only compliance with the functional requirements, but also, to non-functional requirements currently peculiar to space applications, pushing forward these non-functional requirements pointing to the larger flexibility provided by heterogeneous systems.

The Case Study of TAS-E is mainly focalized in including multicore architectures capable of in-flight reconfiguration in actual payload data processing equipment, both for video processing and for telecommunication regenerative payloads. The target is to replace legacy designs in actual flight missions using multicore improved performances to overcome the limitations imposed by classic ASIC designs. To achieve this, TAS-E will define the requirements derived from actual mission scenarios in terms of performances and certification needs and will support the architecture definition and validation activities. Once selected the architectures, TAS-E will implement them in the available processing modules based on the multicore elements both HW and SW.

The different elements developed in the technical tasks will be implemented in the corresponding test benches, one for SSDP and other for Reconfigurable FPGA, implementing the proposed architectures and certification procedures. These test benches will be designed to reproduce as close as possible the actual environmental and operational conditions that will be found in an actual commercial space project to guarantee representatively. The proposed certification techniques and procedures will be compared against previous solutions in the space domain as well as against state of the art solutions in other domains.

2.10.2 Technical description of the case study

This Case Study will be focused on the multicore architectures presently available in the market and the possibilities of implementing them in Space Worth systems that are capable of withstanding the space environment and that can follow the stringent design rules specified for Space equipment. In the same way, in-flight reconfiguration techniques, either by SW modifications for SSDP or by FPGA reconfiguration for Xilinx Virtex5 SIRF, will be covered.

The core of the proposed architectures will be the processor selected by the European Space Agency for the next generation of data handling systems for space applications, i.e. the LEON3 FT which is based on a SPARC-V8 RISC architecture. This processor will be used as base to implement the Scalable Sensor Data Processor Breadboard (SSDP) architecture already under development for ESA to satisfy the needs of the applications that request the fast processing of a high amount of data for smart sensors to be used in future space exploration missions. This architecture combines fixed point DSP IP with a LEON controller. The inherent scalability of the Network-on-chip (NoC) architecture, as well as the efficient combination of GPP and DSP processor cores are very interesting for future large and ultra-powerful processor ASICs, however, a strict validation and certification strategy will be key to allow the widespread usage of such a powerful device in different scenarios with very different criticality constraints.

2.10.3 Case study state of the art

Design of data handling systems and data processing systems for space applications is currently introducing technologies quite new to the space market as multi-core processors or MPSoC. In the space business, the MPSoC are newcomers that are entering the market at an extremely slow speed, especially when compared with the promised advantages that such systems may bring in terms of performances improvement. The main reason for this small adoption ratio is the criticality of the space borne systems and the associated validation and certification procedures. One of the elements blocking this certification is the lack of predictability of critical parameters and therefore lack of deterministic certification tools and procedures. There is a lack of methodologies and tools to support the exploitation of these new technologies in the scope of systems which are compliant to the strict requirements of power consumption, performance under critical conditions, safety, timeliness, security and reliability peculiar to the space applications.

2.10.4 Case study state of the Practice

Test and validation is the core of the stringent discipline of the Space Domain. However, present T&V procedures follow old standards and are not aligned to face the new challenges risen by new paradigms such as in-flight reconfiguration or MPSoC architectures required for future Scientific missions (Rovers, Planetary probes) and advanced telecommunication payloads. The data processing capacity of Satellite data processors is estimated well above the 10 GFLOPs processing mark for the coming decade. To release the full potential of this power it is necessary to evolve present extensive certification procedures to more intelligent techniques and methodologies able to cope efficiently with multi-parallel threading, mixed criticality SW systems and in flight reconfigurable hardware.

Every time some element needs to be modified during the flight, requires a complete phase of modification and validation, implying validation and certification of every single element or block; even though by itself works fine or has no contact with the modified parts of the project.

Nowadays the whole project must be stopped and has to be restarted and treated as a new one, in terms of certification and revalidation of every single line of code and every procedure. These in in-flight missions make them not reachable. In the present, no in-flight mission has the possibility to be reconfigurable; all expeditions are with frozen code with no chance of modification or corrections.

Regarding safety and security into the case study 10 the next tools and methods are being used:

- **Safety** established development process methodology starting with strict requirements from the clients, consortiums and agencies in reconfigurable in-flight methodologies, finishing with the implementation and the verification and validation.
- **Safety** management tool that assures the completely and whole working throughout the entire lifecycle.
- **Security** development process methodology based on the safety process.
- Systematic **security** methodology to identify and analyse the threats early in the design development phase based on the architecture definitions, including a classification of the threats to risk levels.

2.10.4.1 Workflow

During the development process every single new step achieved will be evaluated in terms of accomplishing the strict standards in the space domain.

Every completed valuable block will be checked and fully validated in terms of security based in safety processes.

2.10.4.2 Assessment

Internally TAS-E has been working and studying the integration of FPGAs inserted in “in-flight” products. No experience from TAS-E and partners in reconfigurable FPGAs during the mission due to the strict requirements specified by the owner and agencies. The reconfigurable FPGAs are still not available due to the space specification, and the impossibility of being re-validated and re-certified with every change.

2.10.4.3 Involved roles

For this case study will work the following people from the partners:

- TAS-E:
  - Technical and CS Leader
  - Technical and WP Leader
  - Project Manager
  - Assurance engineer
  - Safety engineer
- GMV:
  - Technical consultor & contributor
- TEC:
  - Technical consultor & contributor
- INT:
  - Technical consultor & contributor
- FBK:
  - Technical consultor & contributor

2.10.4.4 Tools and Tool chains

2.10.4.4.1 Used tools and methods (included guidelines)

All the developments, methods and tools have the specific requirement to accomplish the standards:

- ECSS-E-ST-40C
- ECSS-Q-ST-80C

Which works as guidelines in the whole process, spreading it into different categories:

- Architecture and Modelling
  - Melody Advance
  - Microsoft Visio
- Software Design
  - Enterprise Architect
  - Rhapsody
  - Melody CCM
- Software development
  - Eclipse
- Continuous integration
  - Thales Control
- Source control
  - SVN
  - Git/Stash
- Project and task management
  - Jira
- Requirements
  - Doors

2.10.4.4.2 Tool Chain

Some of the tools used for SW development exchange information, in particular those tools related to the continuous integration and SW validation process.

The traceability of SW development to requirements is currently done manually using Excel spreadsheets.

The output of all these tools is required in order to certificate the SW, from the gathering of requirements to the validation and integration tests before the SW is released to the mission. The ECSS standard defines at what step of the development process each output is required.

2.10.5 Expected technical improvements

The proposed solutions will be used to recode and test the performances of Video Compression algorithms commonly used in the space domain, such as CCSDS122 and 123 which are specialized versions of the JPEG2000 standards. Being a multilayer compression algorithm, it is prone to be parallelized into an MPSoC structure and, as such, AMASS results should show a clear impact on the overall validation and certification of the algorithm. The architecture is prone to parallelization and modularization and as such can be easily linked to in-flight reconfiguration procedures to adapt it to the particularities of the processed images as well as to the evolution of customer needs.

A second application will be data cyphering for on board communications using AES cyphering algorithms presently available. Once more, these algorithms are ready to be parallelized and will show a clear impact of the developments in AMASS.

2.10.5.1 STO1. Architecture-driven Assurance

- Reuse approach of architectural patterns and models on space domain.
- Safety and security of reliable models (timing, efficiency…) with reuse approach to enable cost/time efficient analysis of a system's dependability under consideration of current available process standards.

2.10.5.2 STO2. Multi-concern Assurance

- Creation of an automated Assurance Case to fulfill requirements in an efficient way.
- Objective evaluation of use case functions with focus to functional safety and security.
- Verification and Validation of safety-related and security-related developed measures (methodology, cost efficient test coverage…).

2.10.5.3 STO3. Seamless Interoperability

- Introduction of tool environment to achieve safety and security activates in an economic efficient and consistent way to obey requirements of standards, guaranteeing the validation of independent blocks without the necessity of the whole systems to be revalidated.

2.10.5.4 STO4. Cross/Intra-Domain Reuse

- Objective assessment of security analyzing methods and algorithms. Assure the re-qualification, reuse or enhancement of current methods.

2.10.6 Business needs

Due to the neediness of the everyday more complex space sector, products must be more flexible each day and cheaper. Most of the costs are related to the time involved in the certification and re-evaluation of blocks that have not changed but have been included in a high-level method or algorithm.

With CS10 it is covered the necessity of adaptively basic in such a variable sector in terms of technology and business requirements in the entire time slot (15 years), with reconfigurable in-flight software without the obligation of being re-validated and re-qualified the whole system with every single change.

In economic terms, all these scenarios results will mean for TAS-E:

- AMASS will allow to multiply by 4 the Space Routers accessible market, from 12 to 50M€ accessible per year.
- With present market share, AMASS improvements will allow increasing Space Router Sales from 8M€ to 32M€ by 2020.

2.10.6.1 AMASS Goal 1

G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.

Currently in the space industry, all new designs must go through an exhaustive certification and qualification process which can take years of effort and engineering resources. Reducing this certification time is a key aspect to reduce total cost and overall development time.

The technology developed under the AMASS project will allow to reduce the certification effort by starting the certification process at the system architecture level. This will help reduce the overall assurance and certification effort significantly.

2.10.6.2 AMASS Goal 2

G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.

Another key aspect to reduce the certification effort is by reusing pre-qualified components, or components that have been certified in a previous space mission. Current limitation is the lack of clear guidelines of how pre-qualified components can be reused at the system architecture level. The AMASS project will help to define techniques and methods to improve the reusability of components at the system level, reducing the cost of qualification activities.

2.10.6.3 AMASS Goal 3

G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.

Some of the existing and well-known techniques to improve the performance of HW/SW CPS systems cannot be applied to the space industry due to the lack of the corresponding certification processes or the enormous amount of effort that the qualification would take. A clear example of this limitation is the usage of multiprocessor systems. Although multicore CPUs have been available for many years, its usage in space missions is very restricted due to the difficulties they present from the assurance and qualification point of view. Another clear example is the usage of reconfigurable FPGAs, which is currently restricted in space missions.

The technology developed in the AMASS project will allow the space industry to benefit from the usage of multicore CPUs and reprogrammable FPGAs, increasing the performance, flexibility and reducing the assurance and certification effort.

2.10.6.4 AMASS Goal 4

G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.

N/A

2.10.7 Usage scenarios

Table 30 and Table 31 show the 7 usage scenarios related to Case Study 10.

Table 30. CS10 TAS-E and TEC usage scenarios

ID:	TAS-E UsageScenario 2	TAS-E UsageScenario 3	TEC UsageScenario 16	TEC UsageScenario 17
Related CaseStudy	CS10	CS10	CS10	CS10
Addressed Domains	Space	Space	Space	Space
Scenario Name	SSDP	Reconfigurable FPGA	Model-Based Development	Assurance/Certification Management Tool
Short Description	In-flight SW on MPSoC	In-flight SW on MPSoC Reconfigurable FPGA	1) Support for model-based System, Safety, and Security Engineering 2) Support for Safety and Security Analysis 3) Support for Safety and Security V&V 4) Support for contract-based design for system architecture by safety and security contracts 5) Fault Injection 6) Support for the use of architectural and technology patterns for MPSoC: trade-off based on analysis and certification requirements	Compliance with Standards/ product and process assurance/certification management tool to support the compliance assessment and certification 1) Create semi automated assurance case 2) Address certification issues regarding MPSoC e.g. reconfigurable FPGA 3) Support for Safety and Security Argumentation (GSN)
Stakeholders	System engineer	System Engineer	System engineer Safety engineer Security engineer	Quality Assurance Manager Safety engineer Security engineer
Stakeholder constraints	None	None	None	None
Addressed Business Goals:	G1, G2, G3	G1, G2, G3	G4, G1, G3?	G4
Process Steps	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification -System argumentation	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification -System argumentation	Product development on system level concerning safety/security -System requirements -System design -System analysis -System modelling -System verification -System argumentation	
Concerns	Reliability & performance	Reliability & safety	Safety, Security, Reliability, Availability, Maintainability	Safety, Security, Reliability, Availability, Maintainability
Cross-system certification			No	No
Cross-domain certification	Space & Avionics	Space & Avionics	No	No
Engineering Environment (Interoperability)	• Architecture and Modelling: o Melody Advance o Microsoft Visio • Software Design: o Enterprise Architect o Rhapsody o Melody CCM • Software development: o Eclipse • Continuous integration: o Thales Control • Source control: o SVN o Git/Stash • Project and task management: o Jira • Requirements: o Doors	• Architecture and Modelling: o Melody Advance o Microsoft Visio • Software Design: o Enterprise Architect o Rhapsody o Melody CCM • Software development: o Eclipse • Continuous integration: o Thales Control • Source control: o SVN o Git/Stash • Project and task management: o Jira • Requirements: o Doors	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools-Safety/Security Analyses Tool and V&V Tools	Open Source tools AMASS tools when available for use and evaluation
Challenges	N/A	N/A	Safety and Security co- engineering, innovative fail-operational concepts, Safety and Security and Function Availability co-engineering, definition of architectural patterns	Certification challenges w.r.t MPSoC argumentation patterns for fault tolerance, argumentation patterns for specific technologies
Standards	ECSS-E-ST-40C ECSS-Q-ST-80C	ECSS-E-ST-40C ECSS-Q-ST-80C	ECSS-Q-ST-30, ECSS-Q-ST-40, ECSS-Q-ST-80	ECSS-Q-ST-30, ECSS-Q-ST-40, ECSS-Q-ST-80
Any wishes for usage scenario	N/A	N/A	Reuse of established safety methods for security topic	Reuse of established safety methods for security topic
Any known constraints for usage scenario	N/A	N/A	Not so far	Not so far

Table 31. CS10 INT and RPT usage scenarios

ID:	INT UsageScenario 1	RPT UsageScenario 1	RPT UsageScenario 2
Related CaseStudy	CS4, CS10, CS11	CS10	CS10
Addressed Domains	Space	Space	Space
Scenario Name	INT-US1	N/A	N/A
Short Description	Model-based System, Safety, and Security Engineering Support for Safety and Schedulability Analysis Contract-based Design for System Architecture by Safety and Security Contracts, Contract refinement formal verification	N/A	N/A
Stakeholders	System engineer Safety engineer System and safety engineers collaborate together to definition of requirements and to the building of the architecture. Safety enginner coolaborates with assurance engineer to realize the architecture driven assurance.	Software Engineer Test Engineer	Software EngineerTest Engineer
Stakeholder constraints	N/A	N/A	N/A
Addressed Business Goals:	G1, G2, G3	G4: Reduce certification effort by improving traceability support from HLR to test and SCA results.	G4: Automation of the verification processes (testing, SCA, timing anlysis) within continuous integration systems.
Process Steps	System requirements System design System verification System argumentation	System tests Unit tests	System tests Unit tests
Concerns	Safety	Safety	Safety
Cross-system certification	N/A	N/A	N/A
Cross-domain certification	N/A	N/A	N/A
Engineering Environment (Interoperability)	CHESS and integration with analysis tools. CHESS tool supported activities are: Design, Dependability (usage of MDH tool) and Schedulability Analysis (usage of the MAST tool), Ada Code generation. Integration with OCRA and xSAP fro contracts verification and further dependability analysis support. Use of OpenCert AMASS environment to manage process, evidence, assurance case information.	N/A	N/A
Challenges	N/A	N/A	N/A
Standards	ECSS, SAVOIR-FAIRE	DO178	DO178
Any wishes for usage scenario	N/A	Ada/C/C++	Ada/C/C++
Any known constraints for	N/A	N/A	N/A

2.11 CS11: Design and efficiency assessment of model based Attitude and Orbit Control software development

2.11.1 Short description of the case study

OHB Sweden is responsible for developing the attitude and orbit control subsystem (AOCS) used for a number of different telecommunication satellite platforms. Attitude control is controlling the orientation of the satellite with respect to an inertial frame of reference or other entity. Orbit control is controlling the positioning of the satellite in orbit. Controlling the attitude and orbit requires sensors to measure the satellite orientation, actuators to apply the torques needed to re-orient the satellite to desired attitude and/or orbit and algorithms to command the actuators based on sensor measurements and specification of desired attitude and/or orbit.

The development of critical on-board software applications such as AOCS is continuously becoming more complex as space missions become more autonomous. At the same time, it is expected that the pressure on budget and schedule will continue to increase such that the demand for efficient software development still ensuring dependability and safety will increase.

This Case Study will be based on the AOCS SW developed for the telecommunication satellite platform Electra. Electra is a public-private partnership under the ESA ARTES 33 program serving the purpose of providing the satellite communications industry with innovative products and systems.

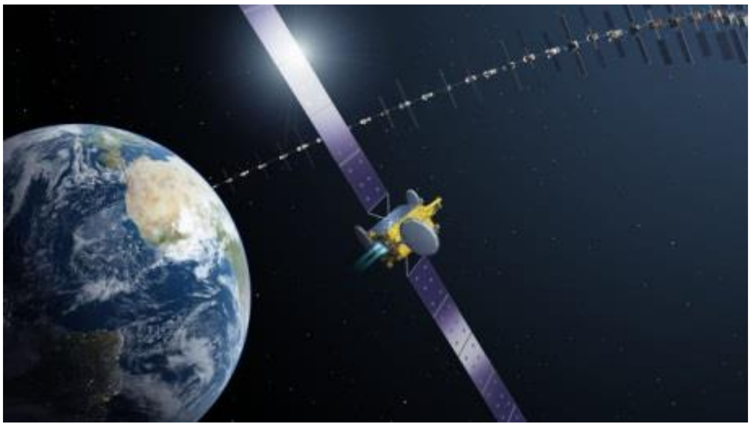


Figure 29. Electra, Telecommunication satellite

The Case Study will exploit the benefits of using the AMASS platform in order to allow for re-use of design, software and process components between different missions with different customer demands. Also, the case study will assess how OSLC can allow for interoperability and seamless integration of model-based system engineering tools with the aim to reduce the cost of assurance activities and evidence management.

2.11.2 Technical description of the case study

This Case Study will focus on selected components of the AOCS SW that will be developed for Electra. The selection of the component(s) shall be based on certain criteria in order to fit the Case Study.

The aim is to identify components that can be reused in different missions. This will require following criteria:

- Generic requirements common for all telecommunication satellite platforms.
- Parameters that can be configured to tailor the components to the requirements of a mission.

The Case Study will assess the effort of developing a reusable component according to ECSS with the help of the AMASS framework.

The Case Study will also assess how OSLC can allow for seamless integration of model-based system engineering tools during the different steps in the development cycle. The development cycle used when developing the AOCS SW for Electra is depictured in Figure 30.

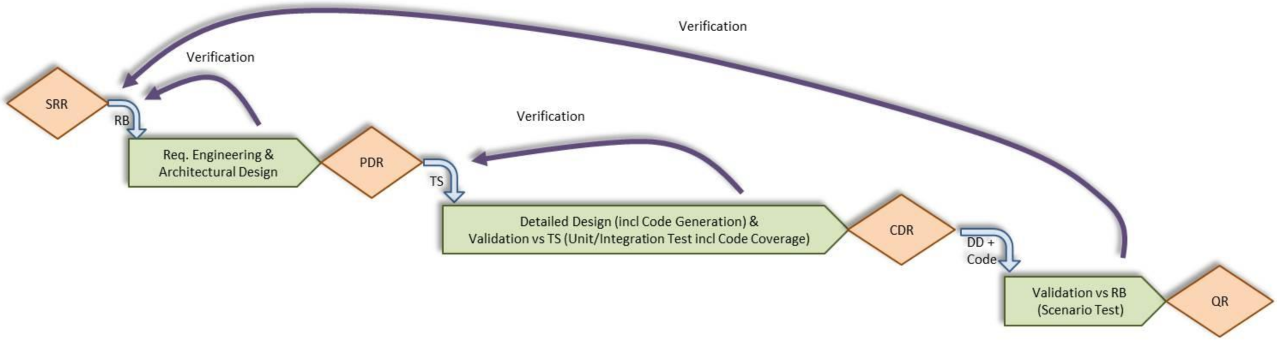

Verification
Verification
Verification
SRR
RB
Req. Engineering & Architectural Design
PDR
TS
Detailed Design (incl Code Generation) & Validation vs TS (Unit/Integration Test incl Code Coverage)
CDR
DD + Code
Validation vs RB (Scenario Test)
QR

Figure 30. AOCS SW Development Cycle

Given the seamless integrated tool-chain, the aim is to automate evidence management performed during verification activities and assess the benefits of the automation.

2.11.3 Case study state of the art

In European space project the development of any SW must be fully compliant to at least the following ECSS standards:

- ECSS-E-ST-40C Software general requirements
- ECSS-Q-ST-80C Software product assurance

There are a number of additional standards for management processes:

- ECSS-M-ST-10C_Rev.1 Project planning and implementation (6March2009)
- ECSS-M-ST-40C_Rev.1 Configuration and information management (6March2009)
- ECSS-M-ST-60C Cost & schedule management (31July2008)
- ECSS-M-ST-80C Risk management (31July2008)

The ECSS also addresses dependability and safety processes on system and software level:

- ECSS-Q-ST-30C Dependability (6March2009)
- ECSS-Q-ST-40C Safety (6March2009)

For critical on-board SW application such as the software associated with AOCS, the mentioned standards require high level of assurance activities and provisions of evidence that SW fulfils system- and ECSS standard requirements.

With model-based design in MATLAB Simulink, a major step has been taken to reduce development time and cost. Model-based design allows developers to prove the design by simulation in MATLAB Simulink before automatically generating the code and avoiding the introduction of manually coded errors while doing so.

One challenge is to avoid re-design of similar components/functions or validation/verification frameworks when creating AOCS SW for different satellite platforms. Another challenge is to semi-automate the assurance activities through seamless integration of the engineering tools used in the development of AOCS SW.

2.11.4 Case study state of the practice

The following methods and tools are already used during the development of the herein described Case Study regarding safety, reliability, availability and maintainability:

- **Requirements engineering and architectural design:** Established model-based design process using DOORS for requirement tracing and Simulink to create and analyse the architectural design.
- **Detailed design (incl. Code Generation & Validation vs TS (Unit/Integration tests incl. Code Coverage):**
  - Established model-based design techniques to implement and simulate the control algorithms in Simulink blocks. Also generation of the code.
  - Implementation of Unit/Integration tests including Code Coverage tests.
- **Validation & Qualification (Validation vs RB):** Scenario testing of the complete AOCS SW with the use of a simulated environment (sensors and actuators, environment and dynamics models).

2.11.4.1 Workflow

This is a description of the workflow of this Case Study and an initial selection of tools.

Table 32. CS11 Workflow

Phase	Activity	Tool
FMECA & FDIR design	Reliability assurance	DOORs
Requirements engineering & architectural design	Specification (Contract based assurance composition)	DOORs & Simulink
	Traceability	DOORs Attribute
Detailed design (incl. Code Generation) & Validation vs TS (Unit/Integration Tests incl. Code Coverage)	Model-based design & Code Generation Code Coverage Test generation.	Simulink Matlab Embedded Coder Gcc, gcov
	Traceability	DOORs Attribute
Validation vs RB	Scenario test using Satsim (Satellite Simulator)	Matlab Satsim
	Traceability	DOORs Attribute

2.11.4.2 Assessment

The results of the Case Study will be assessed as follows:

- Evaluation of the benefits to re-use SW components in accordance with ECSS, by comparing the results obtained following the AMASS project (contract bases assurance) compared to the ad-hoc clone and own method used in the Electra project.
- Evaluation of the benefits to automate evidence management during verification activities, by comparing the results obtained following the AMASS workflow with the results obtained in the Electra project.

2.11.4.3 Involved roles

Roles	Responsibilities
AOCS Engineer	• FMECA & FDIR design • Requirements engineering (AOCS aspects) • Analysis and design of AOCS control algorithms • Validation vs TS • Validation vs RB
Software Engineer	• Requirements engineering (SW aspects)

	• Setup and maintain SW framework • Validation vs TS (Code Coverage)
Quality Assurance Engineer	• Monitors development process to ensure design quality and making sure the SW adheres to standards.

2.11.4.4 Tools and Tool chains

2.11.4.4.1 Tool chains

The table below lists the tools to be used in the Case Study.

Table 33. CS11 Tool Chain

Activity	Tools	Interfacing tools
FMECA & FDIR design	• IBM Rational DOORS	• Matlab • Simulink
Requirements Engineering & Architectural design	• IBM Rational DOORS	• Matlab • Simulink
Detailed Design & Validation vs TS	• Matlab • Simulink	• IBM Rational DOORS
Detailed Design & Validation vs TS	• TSim • Gcc • gcov	• IBM Rational DOORS • Simulink
Validation vs RB	• Matlab • Simulink	• DOORS

The following steps are currently integrated manually in the Electra project:

- Requirements and Design traceability. Currently the developer manually types into DOORS which components the requirements are traced to. Traceability matrices and statistics are generated from DOORS.
- Requirements and Unit/Integration tests traceability. Currently the developer manually types into DOORS which tests validate which requirements. Traceability matrices and statistics are generated from DOORS.
- Implementation of Code Coverage tests are currently performed manually. MATLAB is used to verify code coverage.
- Requirements and Scenario tests traceability. Currently the developer manually types into DOORS which tests validate which requirements. Traceability matrices and statistics are generated from DOORS.

2.11.5 Expected technical improvements

The aim is to increase efficiency and reduce cost through:

- Re-use of methods, components and tests.
- Semi-automation of producing the evidence in relation to the assurance activities.

2.11.5.1 STO1. Architecture-driven Assurance

Contract-Based Assurance Composition for argumentation that the architecture is compliant with the system properties. In case of a system property change event, contract-based assurance composition can also be used for identification of affected architecture/components. Combination of model-based design and contract-based architecture should aim at creating a modular, configurable and re-usable architecture suitable for a general telecom satellite platform.

The AOCS SW is developed using model-based design with MATLAB Simulink.

2.11.5.2 STO2. Multi-concern Assurance

N/A

2.11.5.3 STO3. Seamless Interoperability

Tool Integration Management of development tools (using OSLC and AMASS platform). Create a tool environment to perform semi-automated verification activities in a cost-efficient way to fulfil requirements of standards like ECSS-Q-ST-80C. Since MATLAB Simulink is the tool used for model-based design, it is in our interest to explore the possibility to integrate MATLAB Simulink with a test tool using OSLC with the aim of automatically generating quality assurance evidence.

2.11.5.4 STO4. Cross/Intra-Domain Reuse

Contract-based argumentation for systematic reuse of process and product-based engineering and assurance artifacts. Using the AMASS platform with the aim to understand whether re-use of the assurance assets is possible or determine what further analysis is required to justify compliance.

2.11.6 Business needs

The needs are to find methods and tools to increase efficiency and to reduce cost of developing AOCS SW.

2.11.6.1 AMASS Goal 1

> *G1: to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50%.*

Methods and tools to support argumentation for identification of reusable components and assessments.

2.11.6.2 AMASS Goal 2

> *G2: to demonstrate a potential reuse of assurance results (qualified or certified before), leading to 40% of cost reductions for component/product (re)certification/qualification activities.*

Reuse of components from one mission to another as well as reuse of process-related artefacts from one mission to another.

2.11.6.3 AMASS Goal 3

> *G3: to demonstrate a potential raise of technology innovation led by 35% reduction of assurance and certification/qualification risks of new CPS products.*

N/A

2.11.6.4 AMASS Goal 4

> *G4: to demonstrate a potential sustainable impact in CPS industry by increasing the harmonization and interoperability of assurance and certification/qualification tool technologies by 60%.*

Methodology of seamless integration of the development tool chain (e.g. DOORS, Simulink, Jenkins (or other test tools from AMASS platform), Hansoft and development tools such as MS Visual Studio or Eclipse.

2.11.7 Usage scenarios

Table 34, Table 35 and Table 36 show the 9 usage scenarios related to Case Study 11.

Table 34. CS11 OHB and MDH usage scenarios

ID:	OHB UsageScenario 1	OHB UsageScenario 2	MDHD5UsageScenario2	MDH UsageScenario 3
Related CaseStudy	CS11	CS11	CS11	CS11
Addressed Domains	Space	Space	Space	Space
Scenario Name	Contract based Architectural Design	Seamless integration of tool chain	Product (component) and process reuse	Seamless interoperability
Short Description	Contract-based design for System Architecture by safety, availability and maintainability assurance and argumentation.	Seamless integration of model-based engineering tools for safety and reliability assurance.	Reuse of components from one mission to another as well as reuse of process-related artifacts from mission to another	N/A
Stakeholders	System Engineer - Software Engineer Software Engineer - Assurance Engineer Software Engineer - Test Engineer System Engineer - Test Engineer	System Engineer - Software Engineer Software Engineer - Assurance Engineer Software Engineer - Test Engineer System Engineer - Test Engineer	N/A	N/A
Stakeholder constraints	None	None	N/A	N/A
Addressed Business Goals:	G1, G2, G4	G1, G4	G2	G4
Process Steps	- Requirements Engineering and Architectural Design - Detail Design & Code Generation - Validation vs Technical Specification (Unit/Integration tests incl Code Coverage) - Validation vs Requirement Baseline (Scenario Tests) - Verification	- Requirements Engineering and Architectural Design - Detail Design & Code Generation - Validation vs Technical Specification (Unit/Integration tests incl Code Coverage) - Validation vs Requirement Baseline (Scenario Tests) - Verification	N/A	N/A
Concerns	Safety	Safety	Safety	Safety
Cross-system certification	No	No	Yes	Yes
Cross-domain certification	No	No	No	No
Engineering Environment (Interoperability)	- Requirements Engineering - DOORS - Design & Code Generation - Simulink - Validation vs TS & Code Coverage - Matlab/Alten Code Coverage Generation Tool - Validation vs RB - Matlab & Satsim	- Requirements Engineering - DOORS - Design & Code Generation - Simulink - Validation vs TS & Code Coverage - Matlab/Alten Code Coverage Generation Tool - Validation vs RB - Matlab & Satsim	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools- Safety Analyses Tool and V&V Tools + industry-required tools when appropriate	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools- Safety Analyses Tool and V&V Tools + industry-required tools when appropriate
Challenges	Create contract based method for - re-use argumentation compliant to ECSS. - Consistency analysis	Improve assurance activities through automation of evidence management.	Safety engineering Commonality & Variability systematization	AMASS tools when available for use and evaluation
Standards	ECSS-E-ST-40C ECSS-Q-ST-80C ECSS-Q-ST-30C ECSS-Q-ST-40C ECSS-E-ST-10-02C	ECSS-E-ST-40C ECSS-Q-ST-80C ECSS-Q-ST-30C ECSS-Q-ST-40C ECSS-E-ST-10-02C	ECSS-E-ST-40C ECSS-Q-ST-80C ECSS-Q-ST-30C ECSS-Q-ST-40C ECSS-E-ST-10-02C	ECSS-E-ST-40C ECSS-Q-ST-80C ECSS-Q-ST-30C ECSS-Q-ST-40C ECSS-E-ST-10-02C
Any wishes for usage scenario	-	-	Re-use and automation of verification activities.	Toolinteraction MBSE Tools- Safety Analyses Tool and V&V Tools + Process-modeling tools + industry-required tools when appropriate
Any known constraints for usage scenario	Matlab simulink is a required tool for design, simulations and code generation.	Matlab simulink is a required tool for design, simulations and code generation.	Not so far	Not so far

Table 35. CS11 MDH, FBK and INT usage scenarios

ID:	MDH UsageScenario 4	FBK UsageScenario CS11	INT UsageScenario 1
Related CaseStudy	CS11	CS11	CS4, CS10, CS11
Addressed Domains	Space	Space	Space
Scenario Name	Automation of safety analysis and generation of argument fragments	CS11FBK	INT-US1
Short Description	N/A	1) Modeling of a spacecraft architecture with standard AOCS components and telecommands/telemetry communication 2) Contract-based specification of components 3) Validation of contracts 4) Verification of contract refinement 5) Compare architectures of different missions based on soft requirements and fault trees	Model-based System, Safety, and Security Engineering Support for Safety and Schedulability Analysis Contract-based Design for System Architecture by Safety and Security Contracts, Contract refinement formal verification
Stakeholders	N/A	System engineer, Safety & Security engineer, ModelBased Safety researcher, Verification & validation researcher	System engineer Safety engineer System and safety engineers collaborate together to definition of requirements and to the building of the architecture. Safety enginner coolaborates with assurance engineer to realize the architecture driven assurance.
Stakeholder constraints	N/A	None	N/A
Addressed Business Goals:	G1, G4	G3	G1, G2, G3
Process Steps	N/A	system requirements system design system analysis system modeling system verification evidence for system argumentation	System requirements System design System verification System argumentation
Concerns	Safety	Safety Security Reliability	Safety
Cross-system certification	Yes	N/A	N/A
Cross-domain certification	No	No	N/A
Engineering Environment (Interoperability)	Open Source tools AMASS tools when available for use and evaluation Toolinteraction MBSE Tools- Safety Analyses Tool and V&V Tools + industry-required tools when appropriate	CHESS interacting with analysis tools (OCRA, nuXmv, xSAP) and with OpenCert: Modeling in SySML or AADL using CHESS Formalization using CHESS/OCRA integration Validation and Refinement checked with OCRA Model checking with nuXmv FTA/FMEA with xSAP Collection of evidence for argumentation with OpenCert	CHESS and integration with analysis tools. CHESS tool supported activities are: Design, Dependability (usage of MDH tool) and Schedulability Analysis (usage of the MAST tool), Ada Code generation. Integration with OCRA and xSAP fro contracts verification and further dependability analysis support. Use of OpenCert AMASS environment to manage process, evidence, assurance case information.
Challenges	Re-use and automation of analysis and verification activities.	Reuse of components Comparison of architectures Application of formal methods Generation and reuse of evidence	N/A
Standards	ECSS-E-ST-40C ECSS-Q-ST-80C ECSS-Q-ST-30C ECSS-Q-ST-40C ECSS-E-ST-10-02C	N/A	ECSS, SAVOIR-FAIRE
Any wishes for usage scenario	Re-use and automation of verification activities.	N/A	N/A
Any known constraints for usage scenario	Not so far	N/A	N/A

Table 36. CS11 RPT usage scenarios

ID:	RPT UsageScenario 3	RPT UsageScenario 4
Related CaseStudy	CS11	CS11
Addressed Domains	Space	Space
Scenario Name	N/A	N/A
Short Description	N/A	N/A
Stakeholders	Software EngineerTest Engineer	Software Engineer Test Engineer
Stakeholder constraints	N/A	N/A
Addressed Business Goals:	G4: Reduce certification effort by imp	G4: Automation of the verification processes (testing, SCA, timing anlysis) within continuous integration systems.
Process Steps	System tests Unit tests	System tests Unit tests
Concerns	Safety	Safety
Cross-system certification	N/A	N/A
Cross-domain certification	N/A	N/A
Engineering Environment (Interoperability)	N/A	N/A
Challenges	N/A	N/A
Standards	N/A	N/A
Any wishes for usage scenario	Ada/C/C++	Ada/C/C++
Any known constraints for usage scenario	N/A	N/A

3. Questionnaire Evaluation

The aim of the questionnaire was to get more information about the AMASS project partners and to obtain independent information about the partners. The questionnaire was structured in 5 parts: Part 1 covered general questions; the subsequent parts focused on the development process, the assurance process, and the certification process; finally, Part 5 covered questions related to WP5 regarding tools and evidence management. The results of part 5 are used as input for WP5.

Figure 31 provides an overview about which domains are relevant for the AMASS partners. The three main domains are automotive, rail and avionics, and some partners are working on more than one domain.

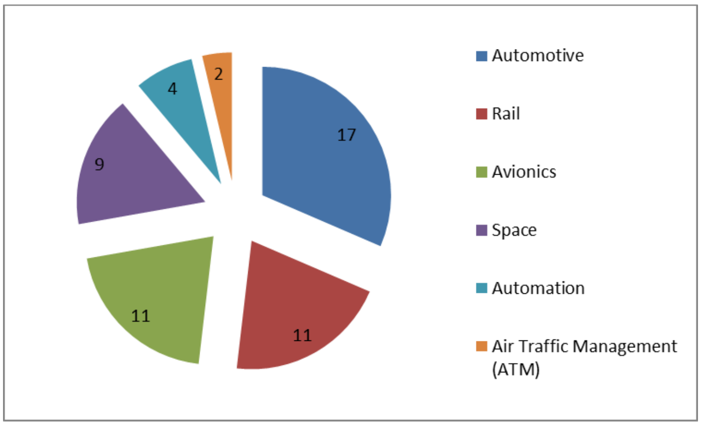


Figure 31. Relevant domains for the AMASS partners

The professional categories of the AMASS partners are present in Figure 32, whereas Figure 33 shows the number of company employees. Industry and scientific are the main professional categories and more than the half of AMASS partners have more than 250 employees.

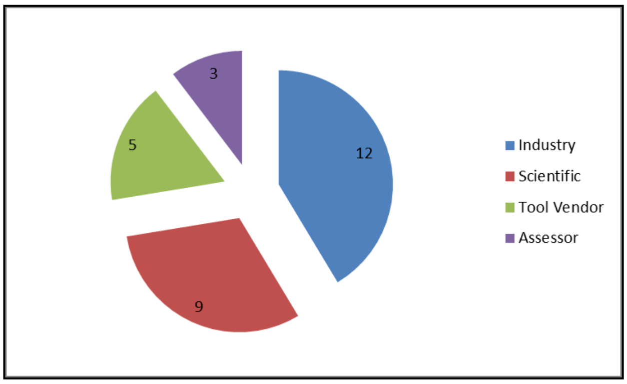


Figure 32. Professional categories of AMASS partners

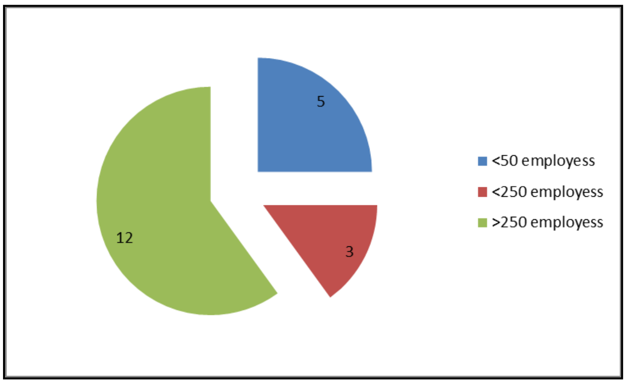


Figure 33. Overview of number employees of AMASS partners

The partners' activities in the development process are shown in Figure 34. Verification and validation are the most frequent activities and the hardware activities are the least usual activities.

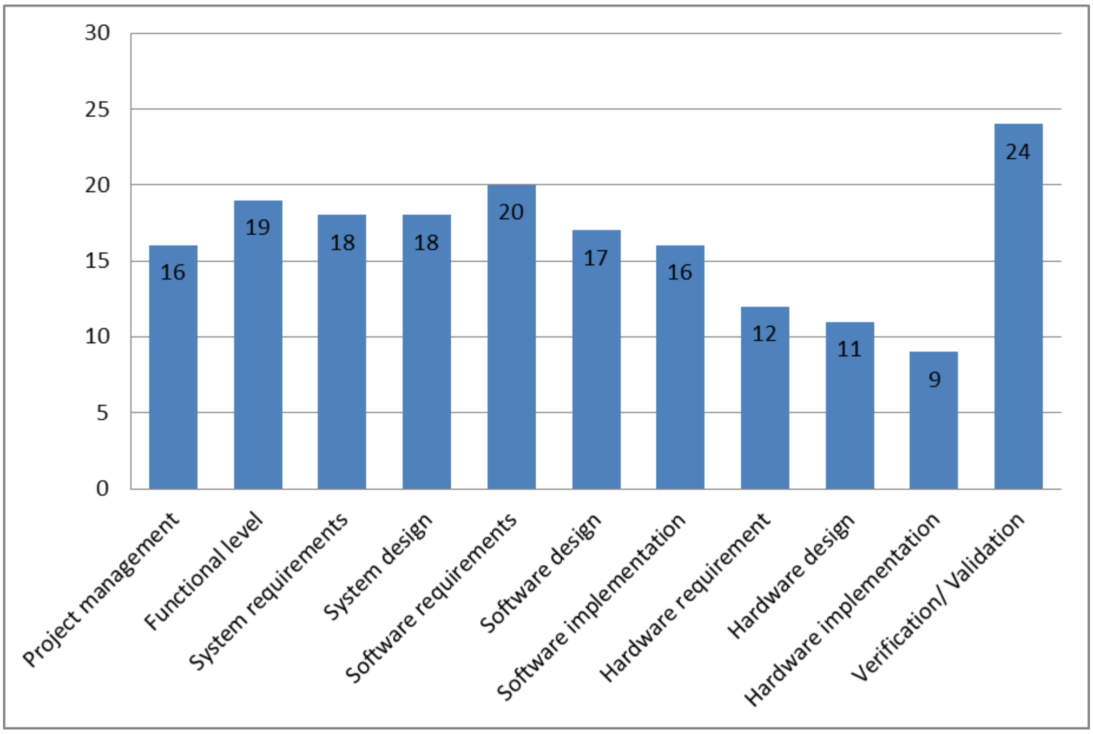


Figure 34. Partners activities in the development process

Figure 35 presents the partners' activities in the assurance process. Most AMASS partners have activities related to safety. The activities related to security are in the contrast less frequent.

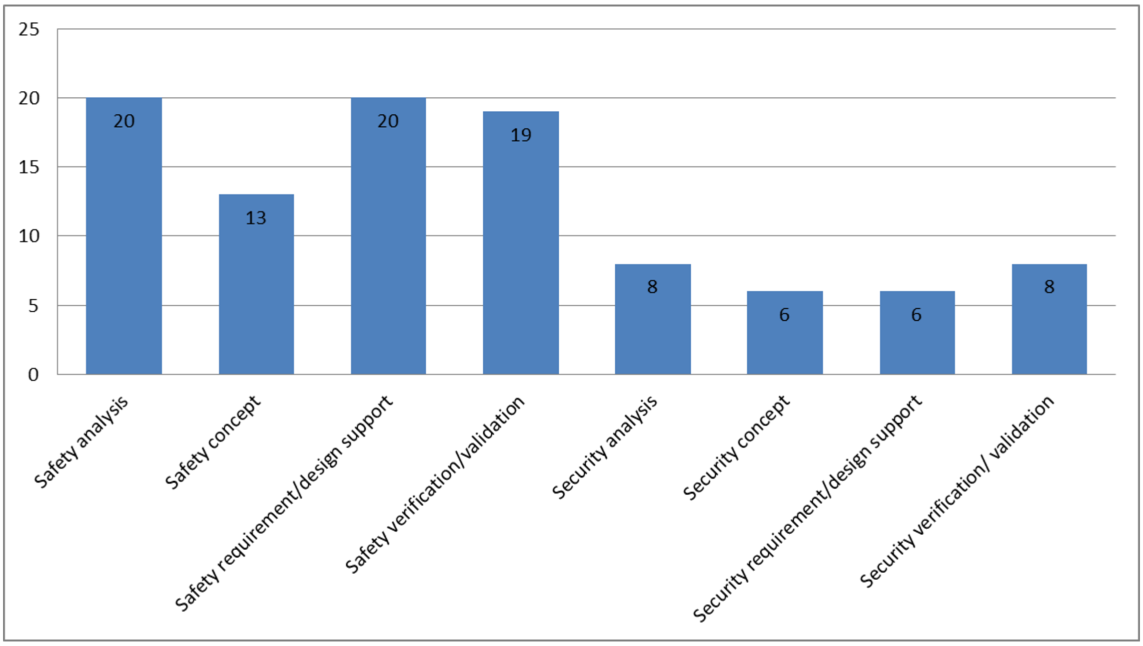


Figure 35. Partners activities in the assurance process

Figure 36 provides an overview of the AMASS partners' activities in the certification process. The activities regarding the assessment in the certification process are the most frequent activities performed by AMASS partners.

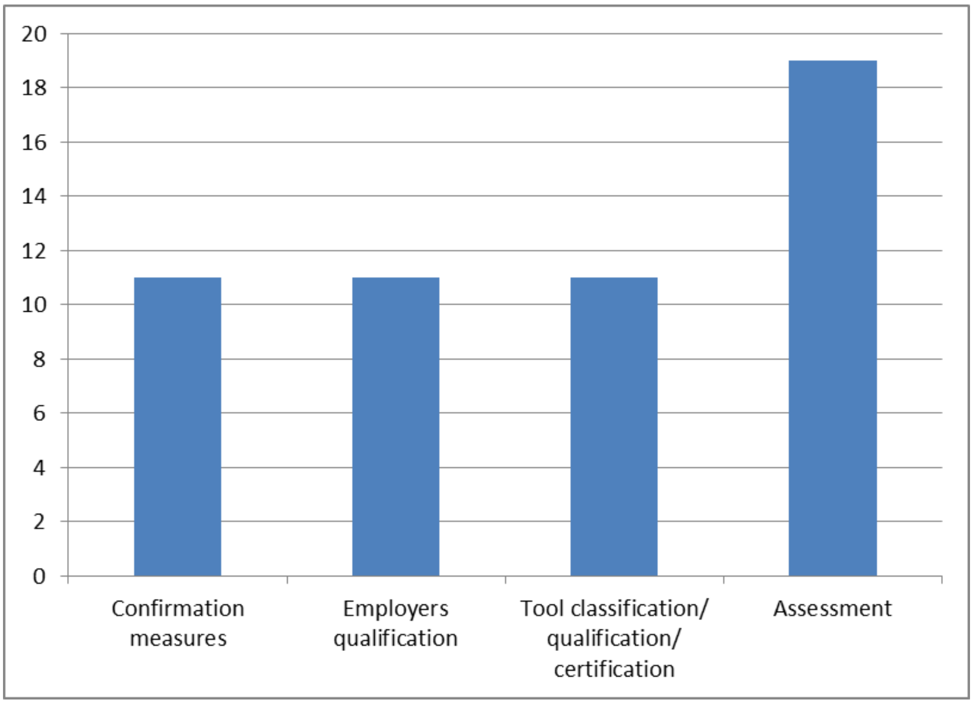


Figure 36. Partners activities in the certification process

4. Conclusions

This deliverable has presented the 11 industrial Case Studies of the AMASS project. For each case study, a technical description of the case study has been provided and the state of the art and the state of the practice regarding the case study have been described. Furthermore, a description of the expected improvements in the case studies by the end of the AMASS project has been provided. Figure 37 shows an overview of which AMASS Scientific and Technical Objectives (STOs) are covered by the AMASS case studies. The evaluation shows that every AMASS STO is covered at least from 8 case studies, which is a promising commitment.

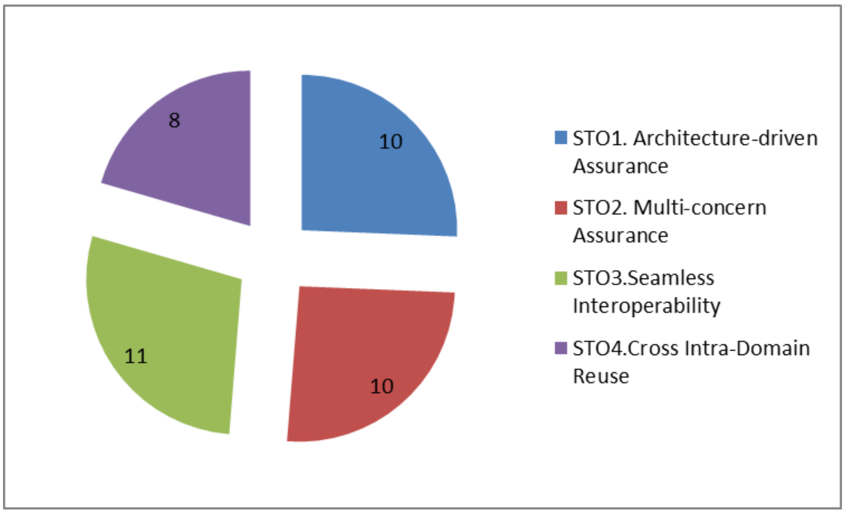


Figure 37. Scientific and Technical Objectives (STOs) covered by AMASS case studies

A description of the specific business needs has been elaborated to improve each of the case studies and show which AMASS goals are covered. The evaluation shows that every AMASS goal is covered by at least 6 case studies (see Figure 38).

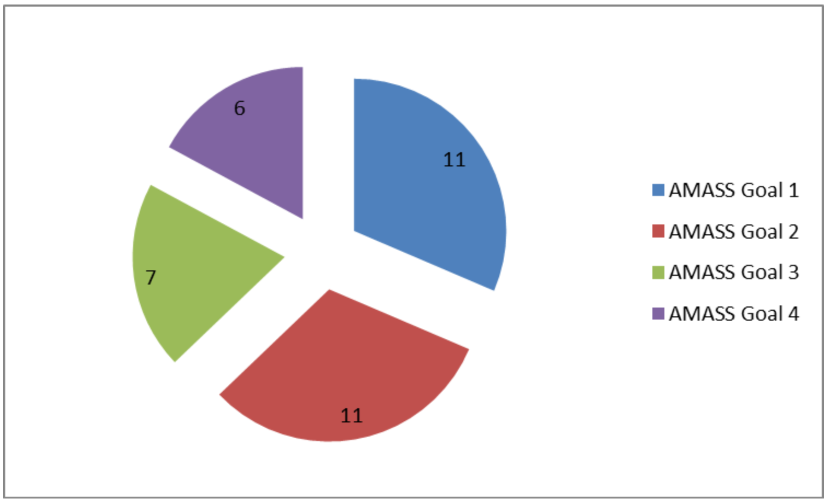


Figure 38. AMASS goals covered by case studies

Finally, the AMASS partners have defined specific usage scenarios for each case study. This has resulted in 87 usage scenarios describing the case study stakeholders and the practices developed by the stakeholders related to those case studies.

Abbreviations and Definitions

ACC	Adaptive Cruise Control
AD	Autonomous Drive
ADAS	Advanced Driver Assistance Systems
AES	Advanced Encryption Standard
AOCS	Attitude and Orbit Control Subsystem
APPSW	Application Software
AR	PSD Authorized Range Area: train position Area where the PSD are authorized to open (decided by the customer)
ARM	Advanced RISC Machine
ASIC	Application-Specific Integrated Circuit
ASIL	Automotive Safety Integrity Level
ATC	Automatic Train Control
ATM	Air Traffic Management
ATO	Automatic Train Operation
ATP	Automatic Train Protection
AUTOSAR	AUTomotive Open System ARchitecture
BCM	Body Control Module
BSW	Basic Software
CACC	Cooperative Adaptive Cruise Control
CAN	Controller Area Network
CAM	Cooperative Awareness Message
CBTC	Communication Based Train Control
CENELEC	European Committee for Electrotechnical Standardization
CMA	Common Mode Analysis
CPS	Cyber Physical System
CS	Case Study
DME	Distance Measuring Equipment
DSP	Digital Signal Processor
DSRC	Dedicated Short-Range Communication
EASA	European Aviation Safety Agency
ECSS	European Cooperation for Space Standardization
ECU	Electronic Control Unit
EEPROM	Electrically Erasable Programmable Read-Only Memory
EMC	Electromagnetic Compatibility
EPCIP	European Programme for Critical Infrastructure Protection
ERTMS	European Rail Traffic Management System
ESA	European Space Agency
FAA	Federal Aviation Administration
FMEA	Failure Modes and Effects Analyses
FMEDA	Failure Modes, Effects and Diagnostic Analysis
FPA	Focal Plane Assembly
FPGA	Field Programmable Gate Array
FTA	Fault Tree Analysis
FTTI	Fault Tolerant Time Interval
GASC	Generic Application Safety Case
GNSS	Global Navigation Satellite Systems
GPP	General-Purpose Pre-processor
GPS	Global Positioning System
HAM	Honeywell Autocode Manager

HARA	Hazard Analysis and Risk Assessment
HCU	Hybrid Powertrain Controller
HMI	Human Machine Interface
HRT	Hard-Real Time
HSM	Hardware Security Modules
HVAC	High Voltage Air Conditioning
IACS	Industrial and Automation Control Systems
ICAO	International Civil Aviation Organization
ICM	Instrument Control Module
IED	Intelligent Electronic Device
IMA	Integrated Modular Avionics
IMU	Inertial Measurement Unit
ISA	Independent Safety Assessor
ITS	Intelligent Transport System
IXL	Interlocking
JAA	Joint Aviation Authorities
LAN	Local Area Network
LCD	Liquid Crystal Display
MBSA	Models-Based Safety Assessment
MCU	Multiple Control Unit
MIT	Module Inspection Test
MPSoC	Multiprocessor System-on-Chip
MTBF	Mean Time Between Failures
NoC	Network-on-Chip
OBSW	On Board Software
OBT	On-board Time
OEM	Original Equipment Manufacturer
OEU	OLCI Electronics Unit
OLCI	Ocean & Land Colour Instrument
OTA	Over-The-Air
PDC	Park Distance Control
PROM	Programmable Read-only Memory
PSD	Platform Screen Doors
RAMS	Reliability, Availability, Maintainability, and Safety
RISC	Reduced Instruction Set Computer
RTK	Real Time Kinematic
RTU	Remote Terminal Units
SASC	Specific Application Safety Case
SCADA	Supervisory Control And Data Acquisition
SCAR	Safety Critical Analysis Report
SEE	Single Event Effects
SEooC	Safety Element out of Context
SIL	Safety Integrity Level
SOC	State Of Charge
SPAR	Safety PSD Authorized Range Area: train position Area where the PSD opening is not dangerous, because the train protects the track side falling. (The SPAR is bigger or equal to the PAR).
SRAM	Static Random Access Memory
SRS	Software Requirements Specification
SSDP	Scalable Sensor Data Processor Breadboard
STO	Scientific and Technical Objective
SysML	Systems Modelling Language
TARA	Threat Assessment and Remediation Analysis

TCU	Telematic Control Unit
UML	Unified Modelling Language
US	Usage Scenarios
TSR	Traffic Sign Recognition
VAM	Video Acquisition Module
V&V	Verification and Validation
ZC	Zone Controller

References

[1] CAR 2 CAR Communication Consortium Manifesto

[2] Design and Experimental Evaluation of CACC

[3] Analysis and design of controllers for cooperative and automated driving

[4] Vehicular-2-X Communication

[5] Handbuch Fahrerassistenzsysteme

[6] New E-Class talks

[7] Highway Pilot Connect

[8] Daimler Mercedes Trucks Highway Pilot Connect - Test Drive

[9] http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Sentinel-3

[10] http://graphics.pixar.com/library/ProtonChi/paper.pdf

[11] Presentation: "System Component Specification (SCS) – use cases for the core prototype", available at https://services-medini.kpit.com/svn/AMASS_collab/WP3/SystemComponentSpecification/SCS_useCases_corePrototype.pptx

[12] Presentation: "Assurance Case Specification – use cases for the core prototype", available at https://services-medini.kpit.com/svn/AMASS_collab/WP4/AssuranceCaseSpecification/UseCases_corePrototype_AssuranceCase.pptx

[13] Presentation: "Evidence Management (EM): Use Cases for the Core Prototype", available at https://services-medini.kpit.com/svn/AMASS_collab/WP5/D5.2_in_progress/Use_Cases/EM_UseCases_CorePrototype.pptx

[14] Presentation: "Compliance Management (CM) – use cases for the core prototype", available at https://services-medini.kpit.com/svn/AMASS_collab/WP6/Compliance%20Management%20Specification/CM_useCases_corePrototype_MDH.pptx .

[15] "Impact of IEC 61508 Standards on Intelligent Electrical Networks and Safety Improvement"; M. Bonnet, M. Laforge, J-B Samuel, Schneider Electric. White paper, 21-Feb-2014

[16] http://spes2020.informatik.tu-muenchen.de/spes_xt-home.html

[17] Ruiz, A., Juez, G., Schleiss, P., & Weiss, G. (2015). A safe generic adaptation mechanism for smart cars. 2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE), 161-171.

[18] Mariani, R., Kuschel, T., & Shigehara, H. (2010). A flexible microcontroller architecture for fail-safe and fail-operational systems.

[19] U.S. Department of Transportation, National Highway Traffic Safety Administration (NHTSA), Assessment of Safety Standards for Automotive Electronic Control Systems, June 2016, https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/812285_electronicsreliabilityreport.pdf

[20] "Out of control: Why control systems go wrong and how to prevent failure"; Health & Safety Executive, UK 2003

[21] Laustsen, B. and Kaiser, B., and Meyer, J.: Modular Replaces Monolithic – New Approaches towards Reusable Safety Concepts. ATZ 2014

[22] Kaiser, B. and Augustin, B. and Baumann, C.: Von der Komponenten- zur Funktionsorientierten Entwicklung in der Funktionalen Sicherheit. 16. Internationaler Kongress Elektronik im Fahrzeug Baden-Baden 2013

[23] The B-Book by J.-R. Abrial. Cambridge University Press, 1996

[24] Atelier B by ClearSy : Parc de la Duranne, 320, avenue Archimède, Les Pléiades III – Bât A, 13857 Aix-en-Provence – France

[25] Systerel Smart Solver by Systerel : Les Portes de l'Arbois, Bâtiment A, 1090 rue René Descartes, 13100 Aix-en-Provence – France

[26] Formal Methods: Practice and Experience by J. Woodcock, P. G. Larsen, J. Bicarregui, J. Fitzgerald. ACM Computing Surveys, Vol. 41, No. 4, October 2009

[27] Formal Methods Case Studies for DO-333 by D. Cofer and S. P. Miller. NASA/CR–2014-218244

[28] Using formal proof and B Method at system level for industrial projects by D. Sabatier. Proceedings of the 1st International Conference on Reliability, Safety and Security of Railway Systems, Paris, 2016, Springer-Verlag

[29] Modelling in Event-B by J.-R. Abrial. Cambridge University Press, 2010