# ESPLANADE

## Efficient and Safe Product Lines of Architectures eNabling Autonomous DrivE

**Public report**

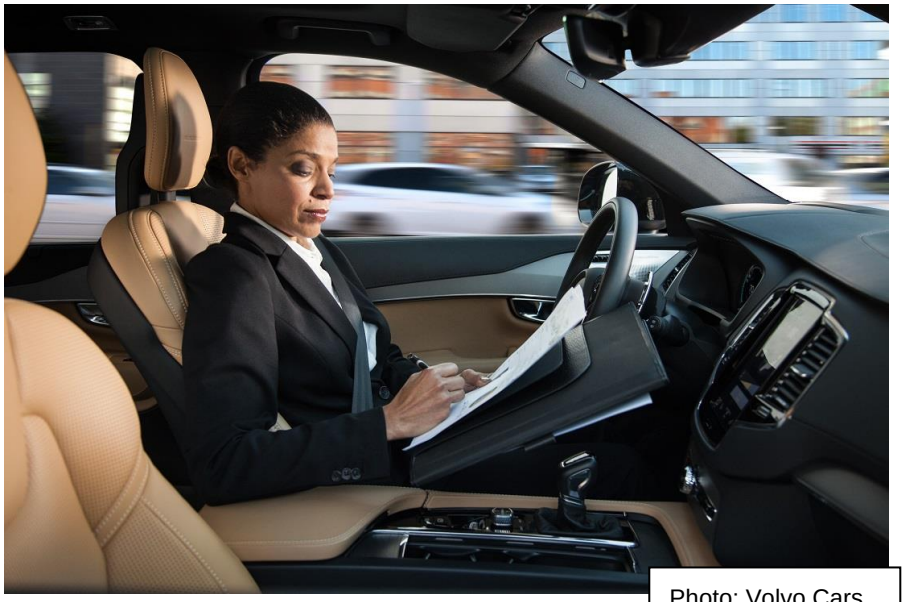

Photo: Volvo Cars


Author: Fredrik Warg (RISE)


Date: 2020-06-11
Project within: Trafiksäkerhet och automatiserade fordon
(Road safety and automated vehicles)



FFI Fordonsstrategisk Forskning och Innovation

# Contents

(Information about FFI in Swedish)

Kort om FFI
FFI är ett samarbete mellan staten och fordonsindustrin om att gemensamt finansiera forsknings- och innovationsaktviteter med fokus på områdena Klimat & Miljö samt Trafiksäkerhet. Satsningen innebär verksamhet för ca 1 miljard kr per år varav de offentliga medlen utgör drygt 400 Mkr.

För närvarande finns fem delprogram; Energi & Miljö, Trafiksäkerhet och automatiserade fordon, Elektronik, mjukvara och kommunikation, Hållbar produktion och Effektiva och uppkopplade transportsystem. Läs mer på www.vinnova.se/ffi.

# 1 Summary

The ESPLANADE project targeted the complex question of showing that an automated road vehicle is safe. This problem is significantly different from safety argumentation for manually driven vehicles. Since the automated driving system (ADS) has complete control of the vehicle when activated, part of its function must be to drive safely. There are several methodological problems that need to be mastered in order to find out how to perform safety argumentation for an ADS. The scope of this project was to provide methods to help solve these problems.

The following topics related to safety assurance of an ADS were investigated:

- How to do safety analysis for Human-ADS interaction?
- How to perform risk assessment and define safety goals (top-level safety requirements)?
- How to determine operational capability and distribute decision in the ADS architecture?
- How to handle incomplete redundancy for sensor systems in the safety argumentation?
- How to ensure completeness and consistency in requirements refinement?

The results include several novel methods as well as new application areas for existing methods.

The ESPLANADE project ran from January 2017 to March 2020 with the partners Aptiv, Comentor, KTH, Qamcom, RISE, Semcon, Systemite, Veoneer, Volvo Cars, Volvo Technology, and Zenuity. 18 scientific papers were produced, of which 16 are at the time of writing published in academic peer-reviewed conferences or journals. Additionally, 13 deliverables in the form of project reports were written.

This final report is a summary of the project results and contains excerpts from the deliverables.

# 2 Sammanfattning på svenska

Projektet ESPLANADE handlade om den komplexa frågan hur man visar att ett automatiserat fordon är säkert. Problemet skiljer sig väsentligt från säkerhetsargumentation för manuellt framförda fordon. Eftersom det automatiserade systemet (ADS) har fullständig kontroll över fordonet när systemet är aktiverat måste en del av dess funktion vara att köra säkert. Det finns flera metodproblem som måste hanteras för att man ska kunna göra en komplett säkerhetsargumentation för ett ADS. Projektets syfte var att tillhandahålla metoder för att hjälpa till att lösa dessa problem.

Följande ämnen relaterade till säkerhetsförsäkring av ett ADS undersöktes:

- Hur gör man säkerhetsanalys för interaktion mellan människa och ADS?
- Hur göra riskbedömning och definiera säkerhetsmål (säkerhetskrav på högsta nivå)?
- Hur bestämmer man operativ kapabilitet och fördelar beslut i ADS-arkitekturen?
- Hur hanteras ofullständig redundans för sensorsystem i säkerhetsargumentet?
- Hur säkerställs fullständighet och konsistens vid kravnedbrytning?

Metoden var att arbeta iterativt genom att titta på alltmer avancerade användningfall och samtidigt förfina metoderna. Utvärdering sker genom exempel och i förekommande fall prototypverktyg. Resultaten inkluderar flera nya metoder samt nya applikationsområden för befintliga metoder. En kort summering är:

- En metod för att klassificera olika typer av interaktion mellan människor och ADS.
- En process för säkerhetsanalys samt designprinciper för interaktionen när en människa överlämnar kontrollen över ett fordon till en ADS eller tvärtom. Processen innehåller existerande metoder som sekvensdiagram, orsak-konsekvensanalys och felträd, men applicerade på människa-maskininteraktion istället för enbart tekniska system.
- Hur man definierar den operativa designdomänen (ODD) för en ADS utgående från önskade användningsfall, vilket innebär en definition av parametrar inom vilka en ADS-funktion är avsedd att fungera, samt strategier för att säkerställa att fordonet håller sig inom sin ODD.
- En metod (kallad QRN) för riskanalys och framtagande av säkerhetsmål. Till skillnad från vanliga riskanalysmetoder bygger den inte på analys av specifika situationer utan på definition av acceptabel frekvens av incidenter med olika allvarlig konsekvens, och en mappning av incidenter till olika klasser av konsekvenser. Säkerhetsmålen uttrycks så att man säkert hamnar inom acceptabla frekvenser.
- Ett ramverk för formell och systematisk hantering av säkerhetskrav med en kombination av åtgärder under utveckling och under drift, bland annat baserat på modeller av osäkerhet.
- Användning av metoden funktionsanalys för att distribuera beslutsfattande på en ADS-arkitektur samt framtagande av säkerhetskrav.
- Säkerhetskontrakt och komponentbaserad design för att underlätta kompletthetsbevisning i kravnedbrytning, möjliggöra kontinuerlig produktuppdatering, samt kunna uttrycka säkerhetskrav för sensorsystem som inkluderar kamera, radar mm.

ESPLANADE-projektet pågick från januari 2017 till mars 2020 med partnerna Aptiv, Comentor, KTH, Qamcom, RISE, Semcon, Systemite, Veoneer, Volvo Cars, Volvo Technology och Zenuity. 18 vetenskapliga artiklar producerades, varav 16 i skrivande stund är publicerade i akademiska konferenser eller tidskrifter. Dessutom har 13 leverabler i form av projektrapporter producerats.

Denna slutrapport är en sammanfattning av projektresultaten och innehåller utdrag ur leverablerna.

# 3 Background

The development of automated driving systems (ADS)[1] has seen major investments in recent years. An ADS is a system capable of performing all of the dynamic driving task of a vehicle for an extended period of time ("driving the vehicle"). The hopes are that such systems will provide more efficient, accessible, and safer transport solutions; but showing that they are in fact safe has been identified as one of the major challenges[2,3].

Before introducing road vehicles with a high level of automation in series production, it is thus essential that they can be proven safe. A key question is what responsibility that is put on the vehicle itself, and what might remain on a manual driver. Full responsibility implies that until a human can be guaranteed to take over the control, the vehicle shall stay responsible to handle the upcoming traffic scenarios. Furthermore, as the ultimate responsibility for a safe behaviour may be transferred back and forth between a manual driver and the vehicle, it is essential that it can be assessed that both parties have a consistent idea of who is currently responsible for the driving task.

Proving that a vehicle equipped with an ADS is safe includes solving both of the following:

- The vehicle always behaves safely when being the responsible party. This includes safe handling of all traffic scenarios possible inside the defined use cases, even if they are not explicitly identified beforehand.
- It is always clear for both the driver and the ADS, which one of them that is responsible. This includes that the chosen procedure for transfer of responsibility must be tolerant to any reasonable human mistake.

To asses that a road vehicle is safely fulfilling what it is promising the driver, is the scope of functional safety. For manually driven road vehicles there is an international agreement in the ISO26262 standard[4], how to argue that the promised vehicle functionalities are safe enough. As there is no human driver that can be found liable if an automated vehicle causes an accident while in control of the driving task, functional safety is a necessary property when arguing that an ADS always behaves safely in traffic. However, as neither the current version of the ISO 26262 standard nor state-of-the-art in research fully answers the question of how to assess that an ADS is safe, there is a need for better methods addressing the characteristic challenges posed by these systems.

# 4 Purpose, research questions and method

## 4.1 Purpose

Without solving the methodologic issues for safety assurance, it will be impossible to commercialize a self-driving vehicle[5]. This project was not about development of functionality for automated vehicles, but rather about methods for analysing and assessing that all implemented functionality is safe. The research questions were, to a large extent, identified in the predecessor project FUSE[6]. The goal is to develop methods that complement existing methods and

[1] *SAE, "SAE J3016 - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," 2018.*
[2] *P. Koopman and M. Wagner, "Challenges in autonomous vehicle testing and validation," SAE International Journal of Transportation Safety, vol. 4, no. 1, pp. 15–24, Apr. 2016.*
[3] *M. Martínez-Díaz and F. Soriguera, "Autonomous vehicles: theoretical and practical challenges," Transportation Research Procedia, vol. 33, pp. 275–282, Jan. 2018.*
[4] *ISO, "International Standard 26262 Road vehicles -- Functional safety," 2018.*
[5] *Markus Hörwick and Karl-Heinz Siedersberger, "Strategy and Architecture of a Safety Concept for Fully Automatic and Autonomous Driving Assistance Systems," IV 2010.*
[6] *https://www.vinnova.se/p/fuse---functional-safety-and-evolvable-architectures-for-autonomy/*

standards, in particular ISO 26262 – functional safety for road vehicles, when applying it to an ADS feature.

## 4.2 Research questions

The research questions were divided into five areas, where a lack of safety assurance methods was identified.

*Driver relations*
In this area the research question was what agreements are needed between the vehicle (ADS) and a human driver in order to be able to argue that the autonomous vehicle is safe, and how do you show that it is safe? The safety argumentation will have to be based on guaranteed limits on human behaviour and performance.

*Hazard analysis and risk assessment (HARA)*
In this area the research questions were how to:

- ensure the HARA is sufficiently elaborated?
- find the hazardous events that create unique safety goals (top-level safety requirements in ISO 26262)?
- create safety goals useful for developing the architecture?
- manage HARA for complex functions and product lines?

*Decision hierarchy and architecture patterns*
In this area the research questions were how to:

- determine in run time the current operational capability with respect to safety integrity levels? This determination shall also be made with the applicable safety integrity level.
- distribute the decisions of the decision hierarchy over the E/E architecture? This also needs to be done for the levels of abstraction of the E/E architecture.

Elaboration: A critical factor for achieving a safe autonomous vehicle, is a proper division of responsibility between levels in the vehicle's decision hierarchy. The general idea is that the tactical decisions shall be aligned to current operational capability.

*Incomplete redundancy for sensor systems*
The research question in this area was to identify a methodology bridging the discrete domain of automotive safety integrity levels (ASIL) used in ISO 26262, and the continuous domain of probabilities. This is necessary when for example evaluating operational capabilities (redundancy and degradation concepts).

Elaboration: A critical factor for getting a safe autonomous vehicle, is a proper division of responsibility between sensors and sensor fusion blocks. In a traditional functional safety methodology whenever redundancy is considered, it is complete redundancy.

*Refinement verification*
The research question in this area was to identify a methodology giving a structure of the refinement verification where each piece of argumentation can be simple enough to be possible to be reviewed by a domain expert.

Elaboration: In each step of safety requirement refinement, it is critical that the resulting composition of refined safety requirements is complete and consistent with respect to the safety requirement. It is considered very hard to formally support such refinement verification because of two major reasons. Firstly, each safety requirement gets part of its semantics from its allocation and only partly from its own expression. Secondly, the semantic gap between the refined and the refining requirements can make them very hard to compare and combine in one

logical argument. When moving from manually driven vehicles to autonomous vehicles, both these reasons imply much more complex consequences.

## 4.3 Method

The research method is to work iteratively with:

- Definition of increasingly more complex ADS use-cases,
- evolving methods addressing the defined research questions in each defined area, and
- validation of methodologies by using relevant examples based on the use-cases as well as tool prototypes.

From a safety responsibility perspective, incremental complexity is defined as more and more ambitious use cases. All use cases include an absolute responsibility for the vehicle as long as we are inside the scope of the use case. This corresponds to a Level 4+ ADS using the SAE taxonomy for driving automation[1]. For such systems, there is no requirement on a human driver to supervise or being prepared to take over at any time.

To support an iterative approach the project was organised in five phases each with a major purpose:

- First phase: Collecting and studying state-of-the-are to prepare to go beyond
- Second phase: Scope confinement and assessing scientific method
- Third phase: Scientific work alignment
- Fourth phase: All scientific papers defined
- Fifth phase: Conclusion

The approach also includes adaption as this is a field with much activity both in research and commercially. The phases were used as a starting point, however in practice there has been an overlap in the phase contents and partly the research questions have been updated during the project as described in Section 6.

To ensure the required expertise to address the research questions the project combines knowledge from Vehicle OEMs, Tier 1 suppliers, expert service companies and research organizations. Furthermore, including two domains (commercial vehicles and cars) was aimed at promoting cross-domain fertilization.

# 5 Objective

The objective of the project is development of methods to reach safe argumentation of automated road vehicles. This includes specific methodology guidelines within five identified areas:

- Methodology for separating the responsibility between the driver and the ADS. This guideline will show how argue that the interaction between humans and the ADS are safe based on human behaviour and performance.
- Methodology for identifying an efficient and still complete set of safety goals even though the ADS has an implicit task to fulfil.
- Methodology to find an efficient E/E architecture and distribute the safety requirements on the elements. This includes finding safe and efficient patterns to implement decision hierarchies and allocate safety requirements on them.
- Methodology to divide safety requirements in an architecture of incomplete redundancy as is typically the case for sensor fusion blocks.
- Methodology for achieving safety requirement refinement verification, especially in situations with large semantic gaps, which are prone to occur when designing autonomous vehicles.

All methods shall furthermore consider a product line approach. This means that they shall result in a manageable complexity even in a context with potentially a large amount of product variants.

During the course of the project, additional questions regarding the development of automated driving systems have emerged both within the project and in the research community as a whole. Hence the project has focused more on some of questions, as described in Section 6. The results are still methods towards reaching a safe argumentation for an ADS, however the current methods should not be seen as complete. Directions for future research is discussed in Section 8.

# 6 Results and goal fulfilment

This chapter summarizes the main results of the project in a number of subchapters each discussing a topic that was dealt with in the project. The chapter concludes with an analysis of how the project has contributed to the overall goals of FFI and specifically to the programme *road safety and automated vehicles*. References to project publications providing more detail of the presented topics are indicated in footnotes.

## 6.1 Use-cases

The project had two use-cases as examples when developing methods for the safety assurance of ADS, however the aim of the methods was to be generic for any type of ADS feature.

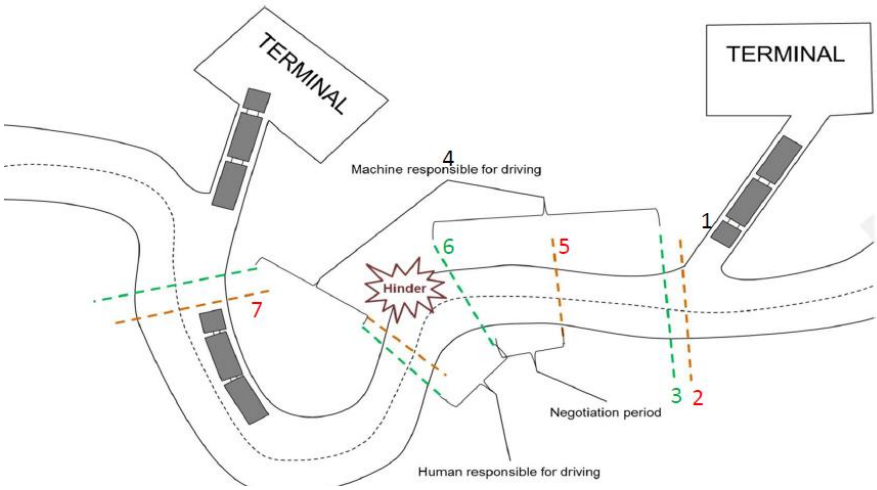


*Figure 1: Example use case for a truck.*

For a truck, the following use-case, illustrated in Figure 1, was formulated:

1. User drives the truck out from a terminal.
2. When the truck is out on highway the user requests the ADS to take over the driving task.
3. When the ADS has judged that it is capable to take over control of the vehicle, it informs the user who then releases control.
4. The ADS operates the vehicle, constantly monitoring the environment to confirm that the vehicle is within, and will remain in, the defined ODD.
5. When the ADS detects an upcoming ending of the ODD, due to road hinders, environmental conditions, etc., negotiation to hand responsibility for the driving back to the user is initiated.
6. If negotiations with the user fails and controls are not taken back, the ADS will decelerate the vehicle to standstill in the current lane.

7. At any time, the user can request taking back control of the vehicle, with negotiation to avoid control misunderstandings. A request to transfer control from ADS to the user can also be initiated by the ADS when following the pre-planned route means leaving the ODD.

For passenger cars, several smaller use-cases/events were instead formulated:

- Activation/deactivation by driver/user
- Deactivation by ADS feature
- Driver/user override
- Sensor blockage
- Autonomous cars on controlled-access highways
- Autonomous cars in urban environments
- Autonomous cars in parking environments

The use cases for passenger cars describe the environmental conditions, the static environment, and the initial state of all involved actors, immediately prior to the scenario described. In this environment, all possible scenarios and/or conflict situations originating from the said initial state are covered by the use case.

More details for the use-cases are in the project deliverables[7,8].

## 6.2 Safe transitions of responsibility

This work focuses on the interaction between a human user and the ADS in level 4 and 5 automated driving. This includes the possible mismatches between the actors in such a setting and usage and discusses methods of identifying and mitigating these to argue safety within the ODDs specified.

*Agreements*

The introduction ADSs puts requirements on achieving completeness of all the agreements, implicit and explicit, entered by the ADS and human stakeholders in various situations. For example, an agreement that is extensively debated, and of great importance, is the transition of responsibility of control between a manual driver and the ADS. Who has the control responsibility shall be unequivocal clear. This type of agreement is a subclass of all the agreements between the ADS and different stakeholders. A way of categorizing all agreements that is needed to be considered during the lifetime of an ADS in a systematic way is shown in Figure 2. In this categorization the agreements[9] are divided into the dimensions of: *stakeholders* – user/passenger, persons in the close proximity of an ADS-equipped vehicle, and society at large; *time scale* – operational (short time frame, one decision point), tactical (medium time frame, more than one decision point and actions) and strategic (agreements for a trip or over the life-time of a vehicle); and safety attributes – whether potential consequences are dangerous (pertaining to functional safety), illegal (pertaining to traffic law) or improper (pertaining to accepted behaviour or customs). These agreements can furthermore be affected by the ADS type (e.g. ERTRAC[10] roadmaps define automated passenger car, freight vehicles and urban mobility vehicles). Figure 2 also shows some examples of agreements and their relation to these dimensions. The focus in the project has mainly been on the agreement *driving control transition*.

[7] *ESPLANADE Deliverable R1v5 - Use Cases for Autonomous Passenger Cars, 2020.*
[8] *ESPLANADE Deliverable R2v5 - Use Cases for Autonomous Trucks, 2020*
[9] *M. Skoglund, F. Warg, and B. Sangchoolie, "Agreements of an automated driving system," in 37th International Conference on Computer Safety, Reliability, & Security (SAFECOMP 2018) - Fast Abstract, Vasteras, Sweden, Sep. 2018. (ESPLANADE publication)*
[10] *'ERTRAC, "ERTRAC Roadmaps". [Online]. Available: http://www.ertrac.org/index.php?page=ertrac-roadmap. [Accessed: 04-Apr-2018].*

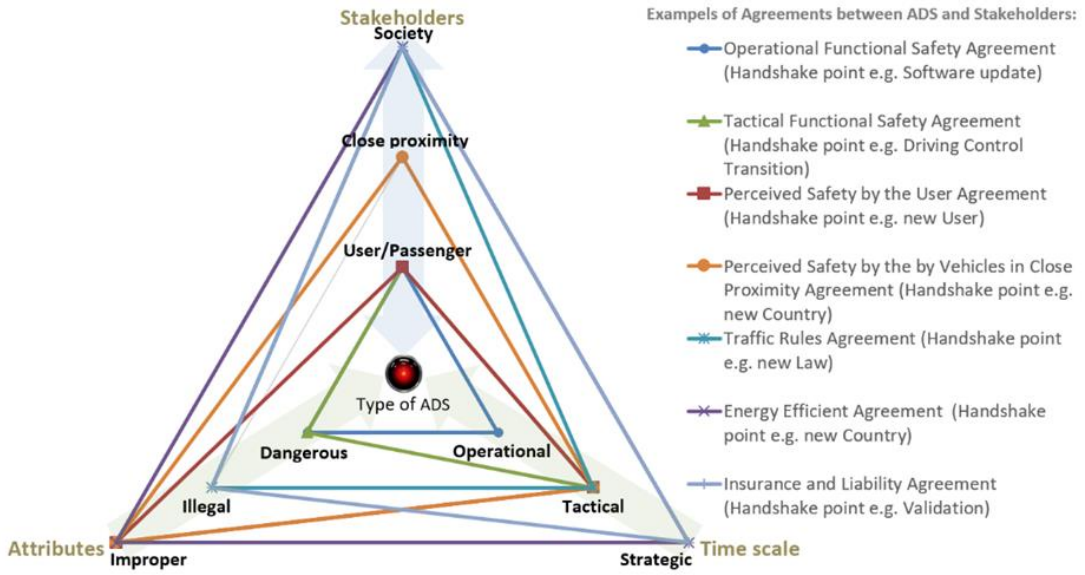


*Figure 2: Agreements between ADS and different stakeholders*

*Transition hazards*[11]

Regarding the responsibility of the driver, the precise L4 definition says that when in charge, the ADS is responsible for the DDT "…without any expectation that a user will respond to a request to intervene".

The introduction of an ADS with full responsibility for the DDT, implies that the problem of traffic safety for an L4-equipped vehicle can be decomposed into three subproblems:

1. Safe driving when the ADS is in charge
2. Safe driving when the driver is in charge
3. Safe transitions between the driver and the ADS.

The first point is obvious when stating that the ADS is responsible for driving and may be the major functional safety challenge for L4-equipped vehicles. The second point includes specific topics coming from the introduction of L4-equipped vehicles. This is because the introduction of an ADS may lead to, e.g., that the driver by mistake relies on an inactive ADS. For the third point, having functionally safe transitions implies showing absence of malfunctions in the ADS transition functionality that may lead to unacceptable risks. This includes risks related to the interaction between the driver and the ADS. For transition protocols there are at least three types of hazards to consider: mode confusion is a situation where the Human User (HU) and the ADS do not share belief of who is performing the DDT; unfair transition is a hazard where either ADS or human is forced to take control in a situation where they are not prepared and able to drive; and stuck in transition means either part is unsuccessful in completing a transition for such an extended period of time that the driving capability is impaired. These transition hazards are illustrated in Figure 3.

[11] *R. Johansson, J. Nilsson, and A. Larsson, "Safe transitions between a driver and an automated driving system," International Journal on Advances in Systems and Measurements, vol. 10, no. 3-4, 2017. (ESPLANADE publication)*

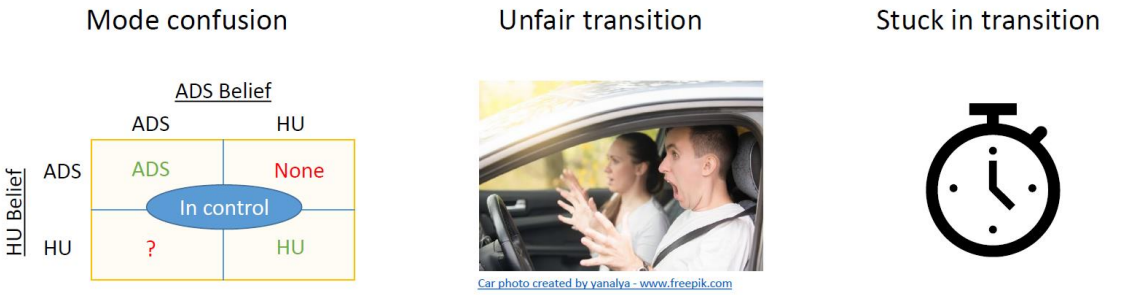


*Figure 3: The three transition hazards identified and analysed in the project.*

By bridging the gap between functional safety principles and practices in HMI design, it will be possible to assess risks that would be found in a human/automation joint cognitive system such as highly automated driving, and design an appropriate system to reduce these risks, achieving safe transitions between the driver and the ADS for a Level 4-equipped vehicle[12].

*Safety analysis of a transition protocol*

The method proposed in the project is safety analysis of a proposed transition protocol with respect to its sensitivity to human error, vehicle component failure, or a combination of these[11]. The analysis process shown in Figure 4 consists of four steps:

1. Propose a transition protocol.
2. Create the interaction sequence with HU and ADS as two communicating entities through the HMI considering the possible combinations of time intervals.
3. Perform cause-consequence analysis (CCA) by constructing cause-consequence diagrams (CCD) based on the interaction sequences, and for each failed event on the CCD perform a fault tree analysis (FTA) considering a model of human behaviour.
4. Perform a risk assessment for identified potential faults and improve the HMI design if the residual risk is considered unacceptable.

We also suggest that the results of the analysis can be used as a part of the argument for safety of the ADS, and thus used in the ADS safety case.

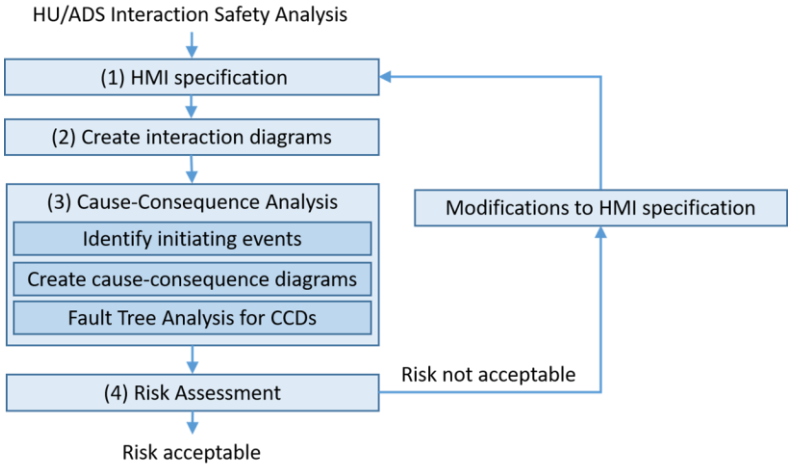


*Figure 4: Process for safety analysis of transition protocols*

[12] *M. Sassman and R. Wiik, "Safer transitions of responsibility for highly automated driving: Designing HMI for transitions with functional safety in mind," in 10th European Congress on Embedded Real Time Software and Systems (ERTS 2020), Toulouse, France, Jan. 2020. (ESPLANADE publication)*

Figure 5 shows an example of one interaction sequence diagram constructed from a proposed protocol, a cause-consequence analysis on one of the initiating events, here defined as points where transition protocol confusion might occur, i.e. there is a mismatch in the beliefs of the ADS and HU about the state of the transition sequence. For each step in the cause-consequence diagram, a fault-tree is constructed to analyse potential component failures and human errors, thus allowing for finding weaknesses in the protocol that may require a redesign.

More details on the transition protocol safety analysis can be found in the project papers “Safe transitions between a driver and an automated driving system”[11] and “Towards Safety Analysis of Interactions Between Human Users and Automated Driving Systems”[13]; while a discussion on how functional safety affects the work of HMI designers can be found in the paper “Safer Transitions of Responsibility for Highly Automated Driving: Designing HMI for Transitions with Functional Safety in Mind”[12].

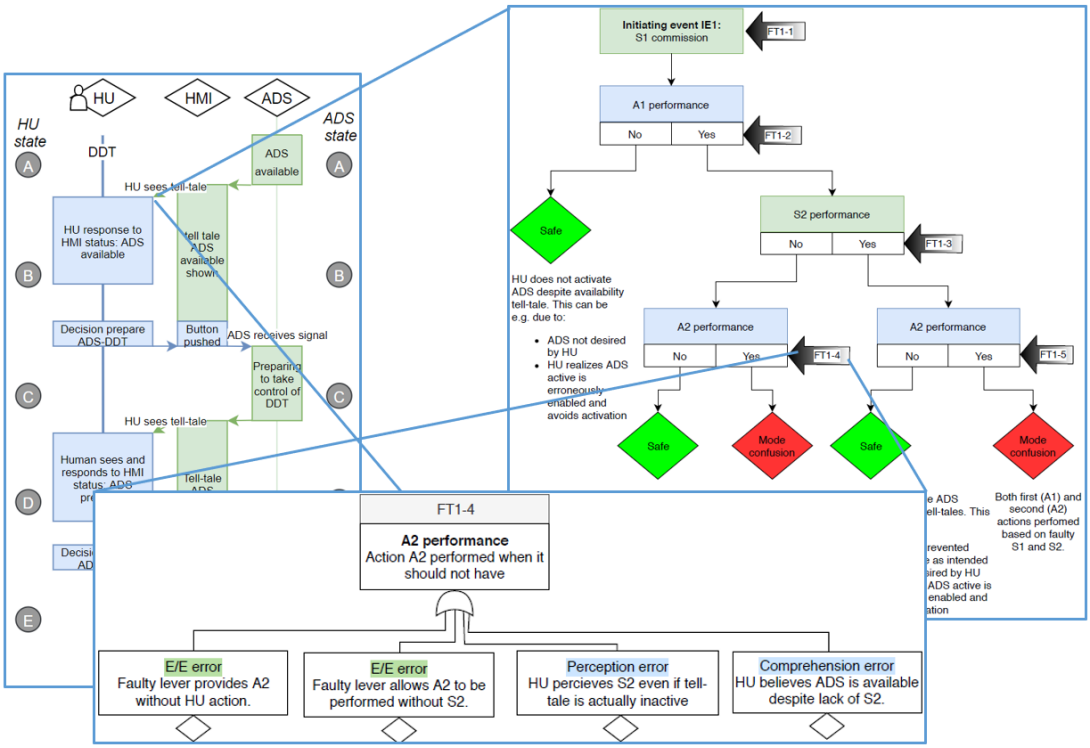


*Figure 5: Cause-consequence diagrams and fault-tree analysis*

## 6.3 The Operational Design Domain (ODD) in safety argumentation

The operational design domain (ODD) of the ADS can be used to restrict where the ADS is valid and thus confine the scope of the safety case as well as the verification. Use cases (UCs) provide a convenient way of providing a strategy for a collection of the operating conditions (OCs) that define the ODD, and further ensures that the ODD allows for operation within the real world. This connection is illustrated in Figure 6. A framework to categorise the OCs of a UC is presented in the project and it is suggested that the ODD is written with this structure in mind to facilitate mapping towards potential UCs. The ODD defines the functional boundary of the system and modelling it with this structure makes it modular and generalisable across different potential UCs. Further, using the ODD to connect the ADS to the UC enables the continuous delivery of the ADS feature.

[13] *Fredrik Warg, Stig Ursing, Martin Kaalhus, and Richard Wiik: Towards Safety Analysis of Interactions Between Human Users and Automated Driving Systems, in ERTS 2020, Jan. 2020. (ESPLANADE publication)*

The pieces of the world model need to exhaustively model the UC and to quantify what the external conditions are around the ADS. If this model is based on a representative set of data, it can be argued that the model itself is adequately complete. To allow the ADS to operate within a UC the ODD needs to encapsulate all the OCs that a such a quantified model of the UC requires.

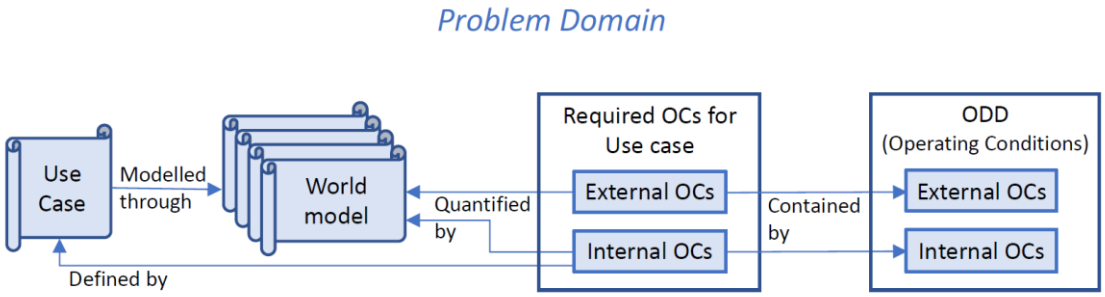


*Figure 6: Structuring the problem domain; use cases and their relation to the ODD.*

Given an ODD, safety requirements to fulfil the safety goals provided in the risk assessment can be defined. The ADS can subsequently be implemented and verified against this set of requirements, as depicted in Figure 7.

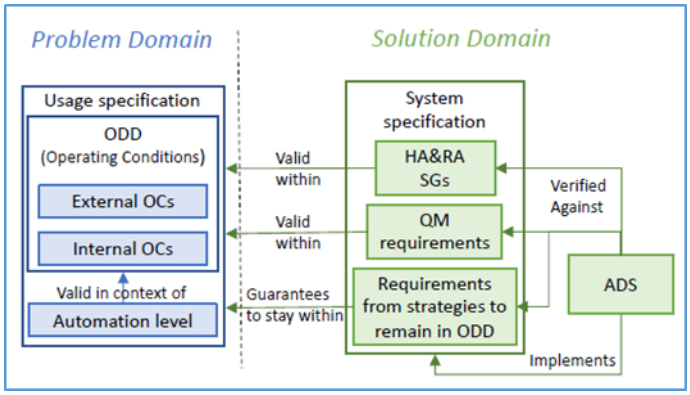


*Figure 7: Connection between ODD in the problem domain and the design in the solution domain.*

*Operating conditions*

For categorizing operating conditions, we use a modified version of the work of Ulbrich et al.[14]. As their framework is geared towards testing and validation, and not modelling and requirement definition, there are some changes that can be made to make it better support the task of categorising the OCs. Our version is shown in Figure 8. There are two main categories of OCs, the internal and the external ones. The internal OCs are the conditions pertaining to the ADS itself and its user. They are defined by the UC directly or they follow from the requirements on the interaction with the user of the ADS. For example, the speed restriction of the ADS (which would be located in Functional Range) is likely given by the UC definition, whereas the time for user to take back control after a request by the ADS (located in Fallback Ready User) can be estimated through user profiling and usage statistics. The external conditions generally need to be modelled and estimated.

[14] *S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, "Defining and substantiating the terms scene, situation, and scenario for automated driving," in Proceedings of IEEE 18th International Conference on Intelligent Transportation Systems, Las Palmas, Spain, Sep. 2015.*

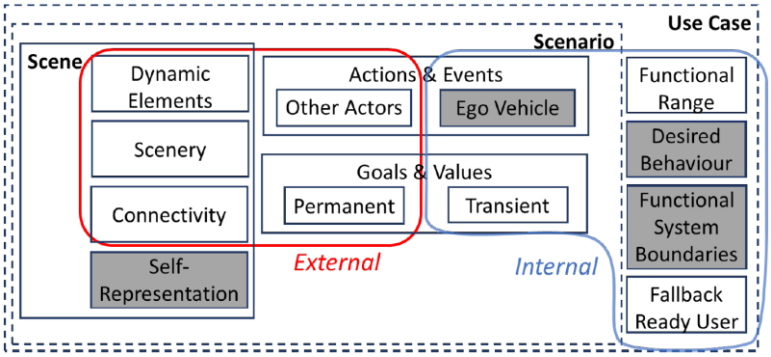


*Figure 8: Categories of operating conditions*

*Strategies for remaining within the ODD*

If the ADS is designed and verified towards an ODD, it is paramount that the it does not leave that ODD. Four generic strategies can be used to ensure this. The strategies are shown in Figure 9. For instance, some conditions, like the internal having a maximum speed of 60 km/h, might be possible to directly guarantee for the ADS as part of the basic feature definition (strategy I), requiring no direct triggering condition at run-time. External conditions may be possible to check while accepting the mission (i.e. as a prerequisite for being able to pass control to the ADS, strategy II), be statically defined (strategy III) or require run-time measuring with sensors (strategy IV).

| Strategies | | | Need to estimate inside ODD in design-time. | Need to define triggering cond. for DDT-fallback | Need for reliable map info. | Need for sensors capable of measuring condition |
|---|---|---|---|---|---|---|
| I | Internal | Inherent in ADS feature definition | N | N | N | N |
| II | External | Checking mission when accepting strategic task | Y | N | Y | N |
| III | | Statically defined, spatial and temporal triggering conditions | Y | Y | Y | N |
| IV | | Run-time measurable triggering cond. related to OC | N | Y | N | Y |

*Figure 9: Strategies to remain within the ODD.*

More details can be found in the paper “Towards an Operational Design Domain That Supports the Safety Argumentation of an Automated Driving System”[15].

## 6.4 Risk assessment for an ADS

Key to ensuring safety is eliciting a complete set of top-level safety requirements (safety goals in ISO 26262). This is typically done with an activity called hazard analysis and risk assessment (HARA). However we argue that the HARA of ISO 26262:2018 is not directly suitable for an ADS, both because the number of relevant operational situations may be vast, and because the ability of the ADS to make decisions in order to reduce risks will affect the analysis of exposure and hazards. This also means that enumeration of operational situations, or relevant scenarios that can occur when using the feature, in the HARA is both intractable and unnecessary. Intractable since the number of potentially relevant operational situations may be vast, making an argument for completeness a very difficult task; and unnecessary since much of this complexity can be confined in the solution domain by means of using tactical decisions and an appropriately defined ODD to reduce risks.

In the project we therefore propose to replace the fixed risk assessment criteria of ISO 26262 with the establishment of a quantitative risk norm (QRN). This norm is essentially a budget of

[15] *M. Gyllenhammar, R. Johansson, F. Warg, D. Chen, H.-M. Heyn, M. Sanfridson, J. Söderberg, A. Thorsén, and S. Ursing, “Towards an operational design domain that supports the safety argumentation of an automated driving system,” in Proceedings of the 10th European Congress on Embedded Real Time Systems (ERTS), Toulouse, France, Jan. 2020. (ESPLANADE publication)*

acceptable frequencies of incidents assigned to a number of *consequence classes* with different severity, where the frequency budget for each consequence class has a strict limit. This norm can encompass both what is traditionally the concern of functional safety (tolerance frequencies related to severity of injuries, compare to the S-factor in ISO 26262) and perceived safety and quality requirements (tolerance frequencies of non-safety related consequences). These consequence classes are depicted in Figure 10 with the safety classes to the right, and the quality classes (grey) to the left, as we would likely accept higher frequencies of quality-related incidents than those which are safety-related.

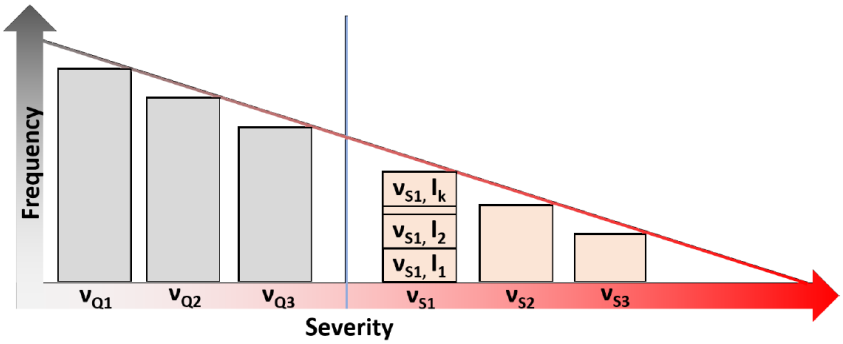


*Figure 10: The quantitative risk norm is based on consequence classes and incident types.*

Having established a QRN that defines what sufficiently safe means in our safety case, the next task is to create safety goals (SGs) that can both be shown to meet the defined limits for all consequence classes of the QRN, and be possible to create a technical solution for. We propose to use classification of incidents into a set of incident types, where each type will result in one SG. If we base our SGs on incident classification rather than a list of hazards and situations, the completeness criteria will apply to this classification. If there exist incidents which are not included in the classification, the safety argument will be flawed. However, we can guarantee completeness by making the classification scheme complete by definition, i.e. every theoretically possible incident belongs to one of the defined incident types. Each type of incident will contribute to one or several of the consequence classes, e.g. if collision between the ego (ADS equipped) vehicle and a vulnerable road user (VRU) is defined as an incident, some of these incidents will lead to a fatality, some to severe injuries, and some to light injuries. As a correctly assigned contribution of incidents to consequence classes is a vital part of the safety argument, it must be well substantiated; however, this is a topic where much data and domain knowledge is available. SGs can then be now formulated for each of the defined incidents.

The need to define operational situations to create safety goals is eliminated with this method, making a completeness guarantee possible. The need to analyse situations/scenarios is confined to the solution domain, which seems appropriate given that what are relevant situations for an ADS is, to a large extent, implementation dependent. Further advantages to the approach are that both safety-related and other unwanted traffic incidents can be included in the same framework; and since the risk norm is decoupled from the implementation the approach is advantageous for product lines since the same risk norm can be used for many variants. I.e., while there may be some variability in the frequency allocation for each incident type (as solutions for variants may have different characteristics) the total acceptable risk for each consequence class will be the same.

A visual summary of the entire process including example incidents and an SG is given in Figure 11.

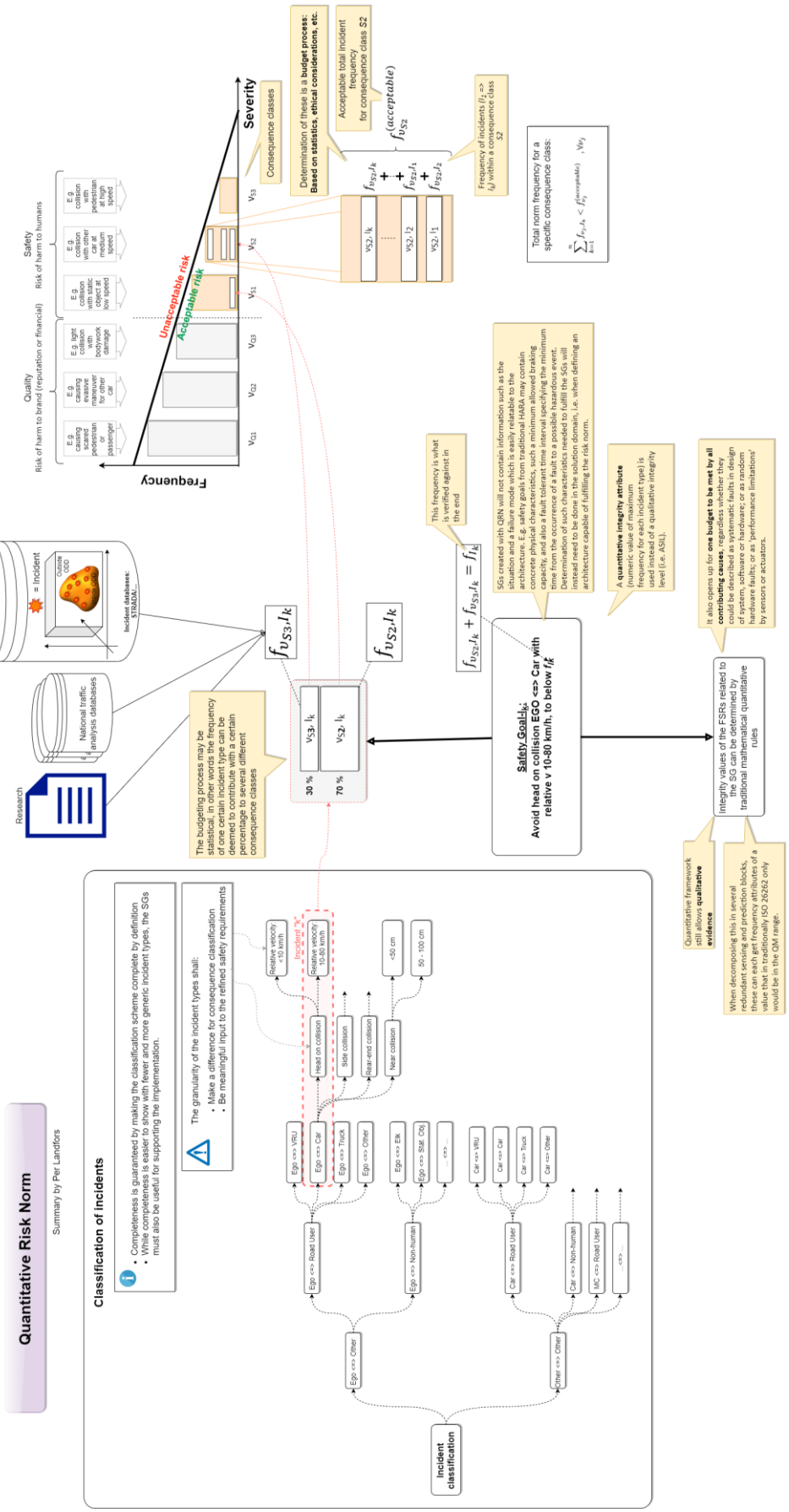


*Figure 11: Visual summary of the QRN approach.*

## 6.5 A formal framework for the reasoning of operational behaviours

Work has been done on a knowledge-base (KB) strategy for a formal and systematic reasoning of safety requirements. On a broader perspective, the work is part of our research on bridging the gaps between intelligent functions, system capability, and dependability for mission-&safety-critical cyber-physical systems (CPS) through a combination of development-time and run-time measures. Further details can be found in the paper ”Design of a Knowledge-Base Strategy for Capability-Aware Treatment of Uncertainties of Automated Driving Systems”[16].

We use the term knowledge-base to refer to the model that stipulates the fundamental facts of a CPS in regard to the overall system *operational states*, *action sequences*, as well as the related *costs* or *constraint factors*. The model constitutes a formal basis for describing, communicating and inferring particular operational truths as well as the belief and knowledge representing the awareness or comprehension of such truths. For reasoning of ADS behaviour and safety risks, each system operational state is explicitly formulated as a conjunction of environmental state and some collective states showing the ADS capabilities for perception, control and actuation. Uncertainty models (UM) are associated as attributes to such state definitions. They describe and quantify the corresponding belief or knowledge status due to the presences of evidences about system performance and deficiencies, etc.

The corresponding modelling support, shown in Figure 12, include *ADS Architecture Model*, *ADS Knowledge Base*, and *ADS Belief&Uncertainty Model*. The key aspect is related to an integrated formal specification of system commitments for automated driving in various operational environments. It is further related to the exploitation of such information, for the design of embedded services, including necessary anomaly detection, and risk mitigation support.

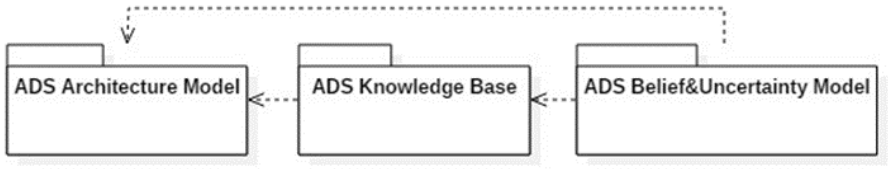


*Figure 12: Modelling support for knowledge-based strategy.*

As the base technologies, the following two existing modelling frameworks are adopted and extended for ADS:

- *EAST-ADL* (Electronics Architecture and Software Technology - Architecture Description Language) for the development of ADS Architecture Model
- *U-Model* (Uncertainty conceptual model for CPS)[17] for the development of ADS Belief&Uncertainty Model. The *ADS Knowledge Base* provides the support for a formal specification of the operational properties across the ADS environment, driver, and vehicle, thereby effectively merging any semantic gaps between the system description and beliefs.

[16] *Chen D., Östberg K., Becker M., Sivencrona H., Warg F., “Design of a Knowledge-Base Strategy for Capability-Aware Treatment of Uncertainties of Automated Driving Systems,” In: Gallina B., Skavhaug A., Schoitsch E., Bitsch F. (eds) Computer Safety, Reliability, and Security. SAFECOMP 2018. Lecture Notes in Computer Science, vol 11094. Springer, Cham, 2018. (ESPLANADE publication)*

[17] *ZHANG, M., SELIC, B., ALI, S., YUE, T., OKARIZ, O., NORGREN, R., “Understanding uncertainty in cyber-physical systems: A conceptual model,” In: European Conference on Modelling Foundations and Applications, pp. 247-264. Springer, Cham, 2016.*

One novelty of this approach is that the introduction of *Uncertainty Models* (UM) constitutes a formal basis for describing and inferring particular operational truths on the basis of formal *Knowledge-Base* (KB). By describing and quantifying the corresponding belief or knowledge status, such models describe the degree of awareness or comprehension of *truths*. The models can be used by system developers for the reasoning of functional and technical commitments at design-time or by embedded services for anomaly detection and risk mitigation at run-time.

The key concepts of integrating *Knowledge-Base* (KB), *U-Model*, and system description in *EAST-ADL* for ADS are shown in Figure 13. Because of this integration, *Evidence* in uncertainty description (e.g. in requirement statements) can have its semantics explicitly given by associated operational behaviour, operation trajectory, or operation performance. Such operations are conducted by system objects given as *EAPrototype*, which is an abstract class in EAST-ADL for the target vehicle or its environment objects and operator. The factors that lead to uncertainty are declared by the associations from *IndeterminacySource* to the EAST-ADL abstract classes for system environment, system functions, hardware components, and system anomaly. With such associations, the sources of non-determinism or indeterminacy are systematically distinguished.

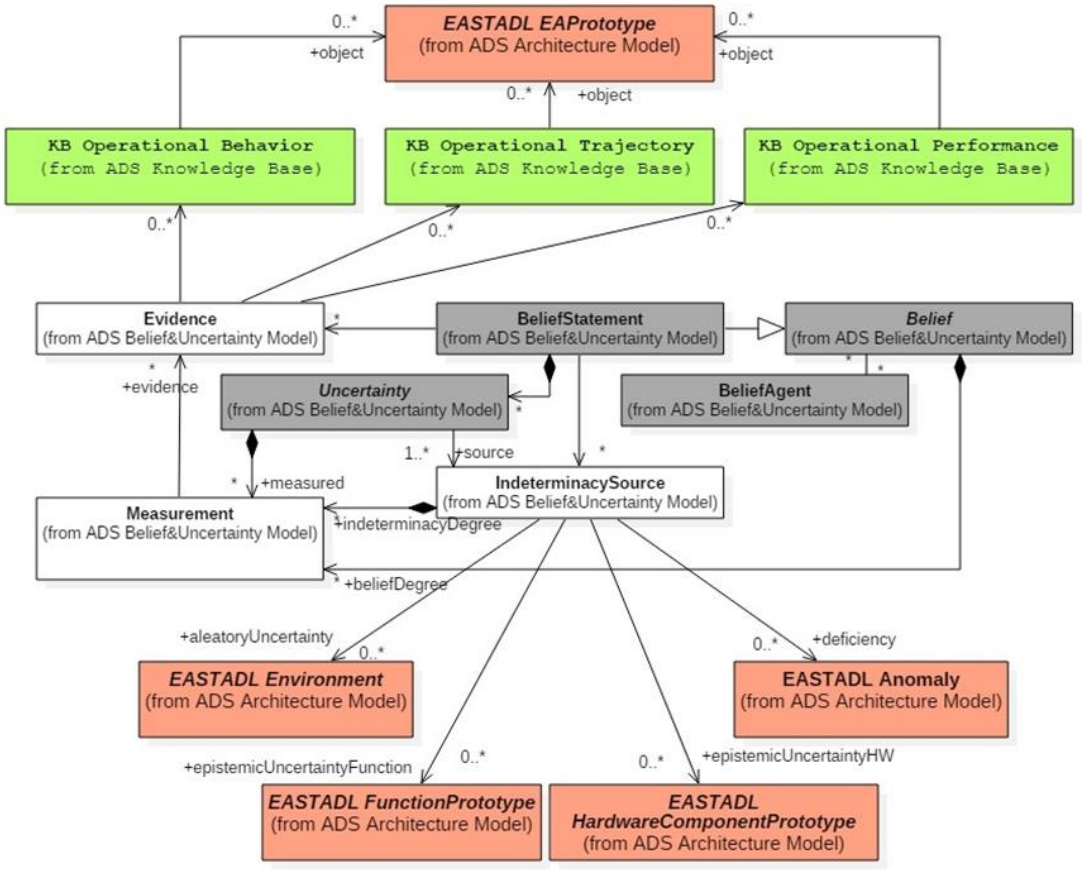


*Figure 13: Key design concepts of integrating UM, KB, and EAST-ADL for ADS description.*

## 6.6 Function analysis for functional architecture development

Function analysis using methods such as FAST (function analysis system technique)[18] was explored as a systematic approach for deriving a functional and logical architecture for an ADS. When systems engineers design new products, Function Analysis is performed to:

- Refine the new product's functional requirements,

[18] *Wixson, J. R., "Function Analysis and Decomposition using Function Analysis Systems Technique". In INCOSE International Symposium (Vol. 9, No. 1, pp. 800-805), 1999.*

- map its functions to elements,
- guarantee that all necessary elements are listed,
- ensure that no unnecessary elements are requested, and
- understand the relationships between the new product's elements.

For an ADS, an architecture can be defined using function analysis together with safety design patterns.

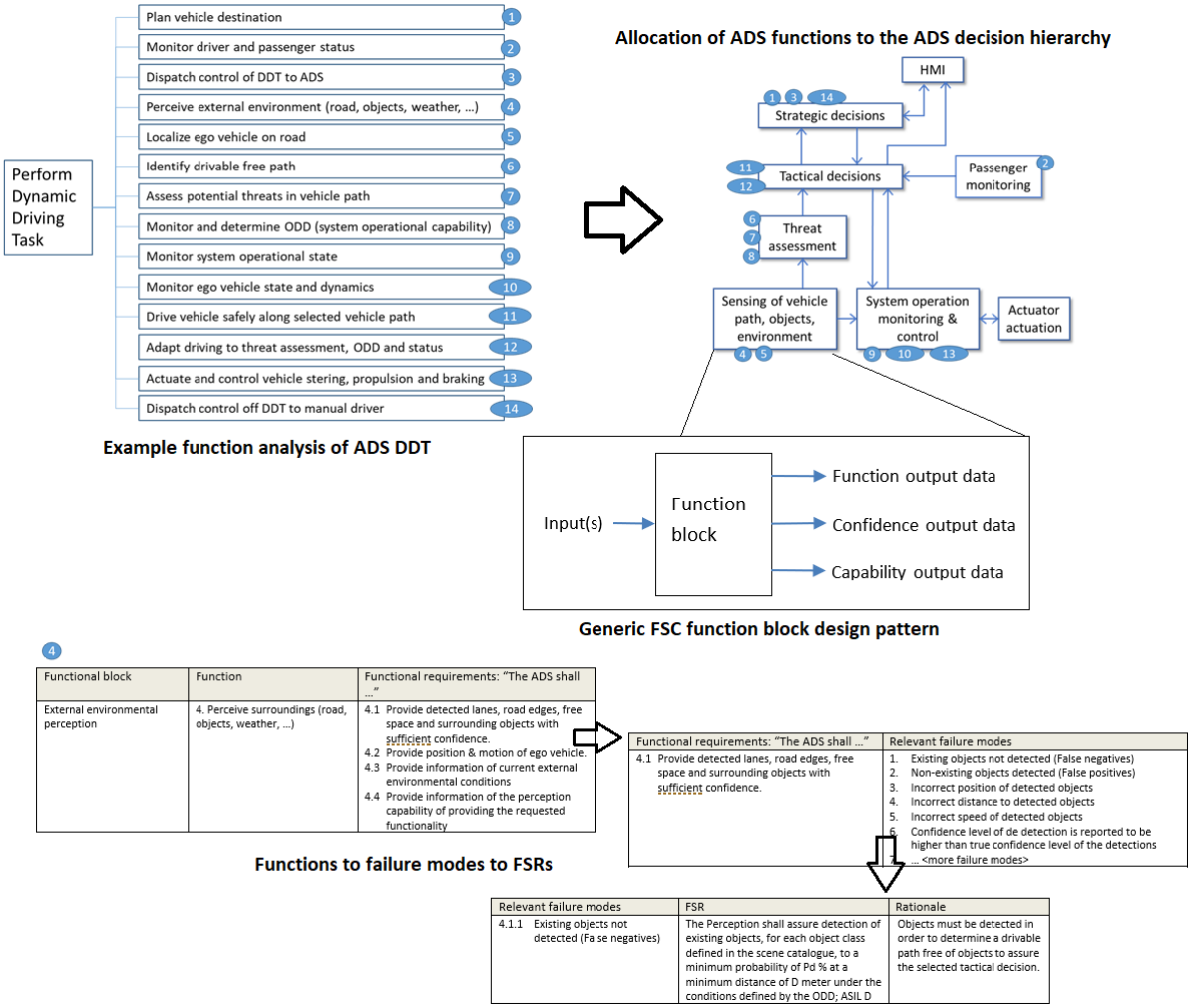


*Figure 14: Example of using function analysis to define an ADS architecture.*

An example of the method is shown in Figure 14 and can be summarized as follows:

- For ADS equipped vehicles, a list of safety goals is defined e.g. with the QRN method. Typically, safety goals imply avoiding collisions, e.g. with various objects in vehicle path such as other vehicles or vulnerable road users.
- In the example the dynamic driving task (DDT) is analysed and broken down to functions in a functional structure by, e.g. applying the FAST method.
- The specific functions are then allocated to the ADS hierarchical model and functional requirements are specified for each of the functions.
- Based on the hierarchical model, a preliminary functional safety concept (FSC) architecture is developed taking different design considerations into account. The FSC specifies functional safety requirements (FSRs) as basis of safety analysis, covering the specific functionality, capability, information and confidence each block must guarantee and the prevention of failure mode propagation. Each functional block of preliminary FSC architecture are preferably described using a generic design pattern as illustrated in the figure.

- A safety analysis is carried out identifying the safety relevant failure modes of the item function and its decomposed functions based on a simplistic error model. The nominal response of each function is defined by the functional requirements. A system fails when characteristics of the requested functionality can't be provided. Failure modes are identified by applying FMEA method using guide words such as omission/commission, too early/late, value too much/little. Faults can be categorized in input faults, internal faults and external noise sources.
- A logical and physical architecture is developed based on the preliminary FSC.

More information and a more detailed example can be found in the project deliverables[19].

## 6.7 Safety contracts for requirements in component-based design

Contract-based design for software development[20] was pioneered in the 1980s. For safety-critical systems, a variant of contract-based design called safety contacts, e.g. as described by Bate et al.[21], has been proposed.

With contract-based design, a component contract specifies some expected functional and technical properties of a system unit given some specific operational contexts. The compositional contexts for such a unit as well as the basis for contract derivation are normally given by the system architecture models. Safety contracts are focused specifically on the properties of a system that may lead to a breach of safety requirements. The key content of a safety contract is therefore related to specification of component fault models and related safety measures. For a component, the fault model describes how a failure arises whenever there is a violation of a contract. Safety measures become necessary for the fault avoidance, fault tolerance and treatment. For example, a safety measure could be related to the conformity control of environmental conditions and component behaviour, stating that *for every occurrence of a 1-bit error on the input of a component, no failure will be produced on the outputs of the same component*. As for a normal contract, the specification relies heavily on the system architecture models that define the mappings of safety requirements and safety measures. With safety contracts, the safety properties of elements can also be matched in an, at least partly, automated way.

In the project, safety-contract based design has been investigated as a method for several purposes; for safety requirements refinement to help show that a refinement is correct and complete, for allocation of safety requirements on sensors, and to allow for an agile way-of-working with continuous deployment and management of product lines.

*Contracts for requirements refinement*
Some systems need to be complex to successfully deliver the service specified (intended). To handle this complexity a top down development strategy is usually applied where the system is described and developed stepwise in different layers of increasing level of detail expressed as requirements and different architecture views. The development starts by defining the highest layer of the system representation which contains the actual product intentions (i.e. functional and non-functional requirements). This highest level is then decomposed into one or several intermediate levels in which the level of detail is increased until the lowest level which is the actual design representation.

[19] *ESPLANADE Deliverable R5.2v4, "Methods for allocating safety requirements on decision elements – project results", 2020.*
[20] *B. Meyer, "Design by contract and the component revolution," in Technology of Object-Oriented Languages and Systems (TOOLS-34"00), 2000.*
[21] *I. Bate, R. Hawkins, and J. McDermid, "A contract-based approach to designing safe systems," in Proceedings of the 8th Australian workshop on Safety critical systems and software-Volume 33. Australian Computer Society, Inc., pp. 25–36, 2003.*

The different abstraction levels of the system representation can be considered as different languages describing the same system. Between representations, there will typically be a semantic gap. The meaning of the term “semantic gap” is the semantic difference between two descriptions of the same object. The more different semantics, the larger semantic gap. This is one of the reasons why intermediate levels of representation are used, i.e. to minimize the semantic gap between the representations. Refinement verification is the activity of verifying that the precision of the system representation is maintained after one subdividing step has been performed, e.g. ISO 26262 requires such an activity. The recommended methods for verification include reviews, safety analysis, and semi-formal verification, but no further guidance is given about how to handle the potential semantic gap. The more complex the system is, the larger is the potential semantic gap between the high-level representation of the product intentions and the low-level design representation of the same system. Hence this is a relevant problem for an ADS which by necessity will be a complex system.

The investigated solution, described in more detail in the project report[22], includes using component-based design, which essentially mean the design will consist entirely of Safety Elements out of Context (SEooC) as discussed in ISO 26262. This means that the component or element is not developed in the context of a particular system or vehicle. Hence the context must be assumed during SEooC development. When the component is deployed in a target, the assumed context must be compared to the actual target context.

The components use safety-contracts and ensuring the integrity of the design thus means matching safety contracts. Components will also have a complete *safety case fragment*, which includes everything from an ISO 26262 safety case that is relevant for the component, such as safety plan; safety work products; and a safety assessment that will be performed on the component. A safety case fragment and how fragments are composed into a complete function (item) is illustrated in Figure 15. Components will exist in all abstraction levels of e.g. the ISO 26262 lifecycle.

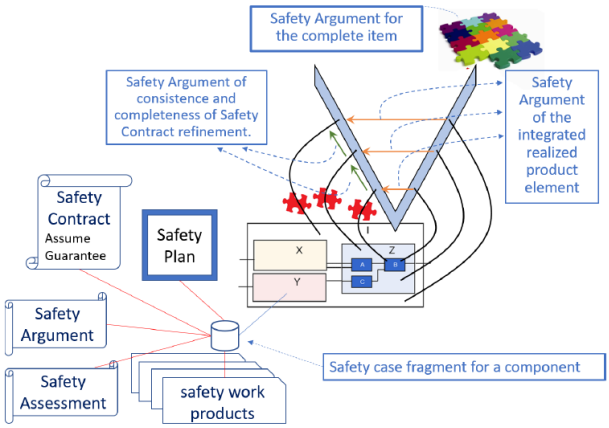


*Figure 15: Safety case fragments and composition into a complete item.*

We illustrate the design of a complete item with the “leaning V”-model shown in Figure 16 (compare with the V-model lifecycle of ISO 26262). It is illustrated according to the dimensions of abstraction and aggregation. It is understood here that verification & validation (V&V) is done in several step of increasing aggregation, i.e. larger and larger blocks with corresponding function are tested: from V&V of units (low aggregation) to V&V of the entire feature (high aggregation). Note that the tested blocks are the “real” parts, units, elements and subsystems.

[22] *ESPLANADE Deliverable R7.2v4 - Methods for safety requirements refinement verification - Project results, 2020.*

On the vertical axis we see high level of abstraction to low levels of abstraction. On the horizontal axis we see low aggregation to high aggregation. Each box in the figure represents a model of the item/function. The different boxes represent this model at different aggregation and abstraction levels. The bidirectional arrows are contracts.

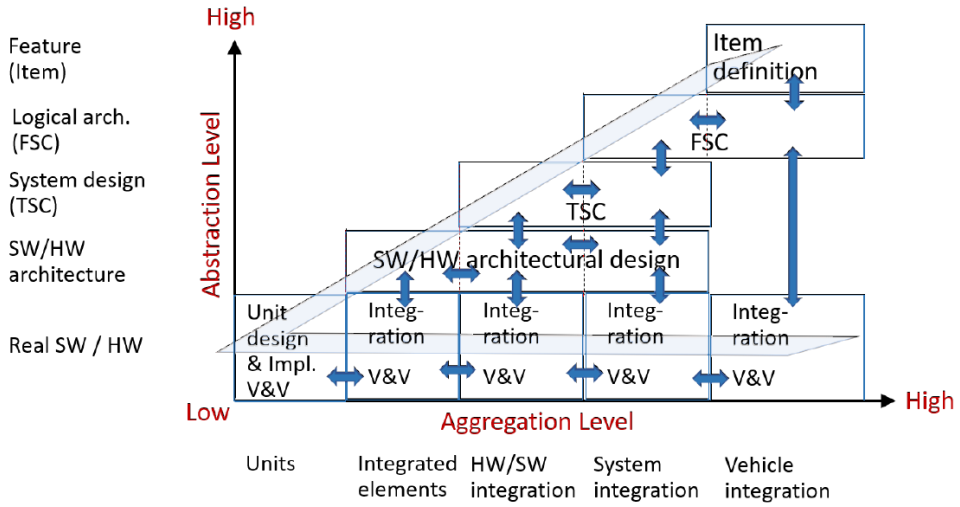


*Figure 16: The "leaning V"-model.*

Contracts by themselves will not solve the semantic gap. The method gives a framework for "taking reasonably small" refinement steps and together with satisfaction arguments[23], the completeness and consistency properties of the contract refinement relation constitutes the refinement requirement verification. By using safety contracts with formal or semiformal specifications, compositions can also be more strictly (or formally) verified. Figure 17 shows a simple semiformal contract example for an ADS feature but finding a way to express contracts which is both efficient and easily usable remains as future research.

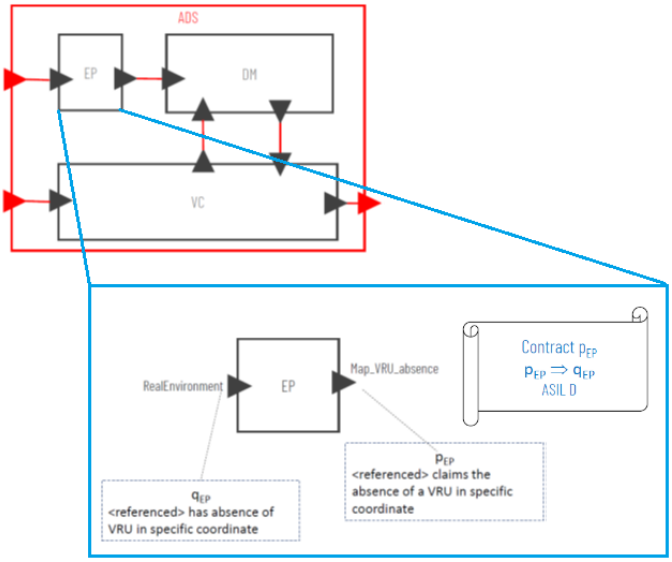


*Figure 17: Example contract part of an ADS on a high abstraction level.*

[23] *Bergenhem, C., Johansson, R., Söderberg, A., Nilsson, J., Tryggvesson, J., Törngren, M., & Ursing, S., "How to reach complete safety requirement refinement for autonomous vehicles". In CARS 2015-Critical Automotive applications: Robustness & Safety, 2015.*

*Contracts for safety requirements on sensors*

Safety contracts together with failure mode analysis has also been investigated specifically for the purpose of assigning safety requirements on sensor components and sensor systems. In the example in Figure 18, from the paper "On Perception Safety Requirements and Multi Sensor Systems for Automated Driving Systems"[24], both functional and safety contracts are defined for each of the function blocks. A proposal is to define safety contracts with assumptions and guarantees for different environmental conditions or operating conditions where the sensors have known limited capability and sensing performance. The guarantee is then assured up to certain ASIL level, depending on the known limitations. The world model must append information of current capability and detection performance for further use in the decision hierarchy.

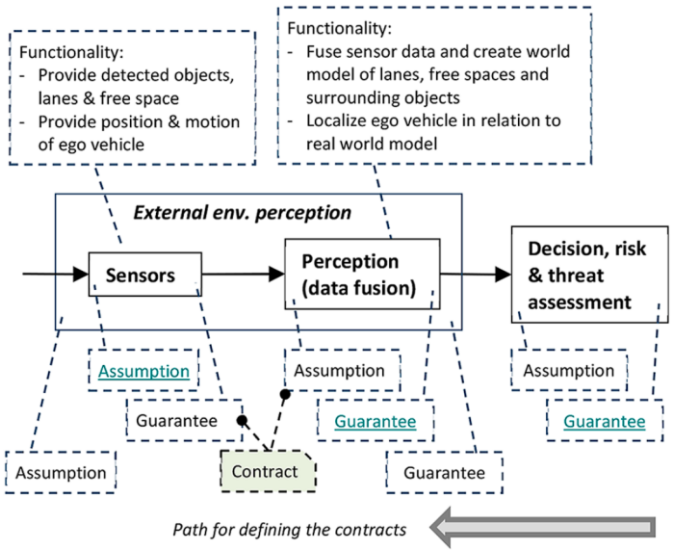


*Figure 18: Safety contracts for sensor systems[24].*

*Contracts for continuous deployment*

Building on the approach described for requirements refinement, the use of safety-contract based design and safety case fragments also has the potential to enable another vital property for the deployment of automated systems: the ability of frequent updates. The traditional way of working with development and assessment of dependable systems is reaching its limits. Driving reasons for this are the emergence of highly complex and quickly evolving dependability-critical systems such as ADSs, and increased connectivity, which necessitates a change in how products are managed, e.g. the possibility to do quick updates to fix security holes becomes necessary. It is also expected that ADS equipped vehicles come in several variants and new versions at a high pace, e.g. to be able to put a product on the market quicker with a more conservative driving style, with the promise to be able to improve performance gradually. For each variant or new version, an updated set of safety cases is needed to show that the safety of the product is maintained.

The complete way-of-working to achieve the ability to perform continuous deployment (i.e. releasing every update to the customer and in a high pace) encompasses these practices:

- The use of component-based design and property-specific contracts (e.g. safety contracts) specifying the components' assumptions on the environment and what it can guarantee provided the assumptions are fulfilled.

[24] *Cassel, Anders, et al. On Perception Safety Requirements and Multi Sensor Systems for Automated Driving Systems. No. 2020-01-0101. SAE Technical Paper, 2020. (ESPLANADE paper)*

- Modular assurance cases where components have their own assurance case fragments relevant to the component context, facilitating reuse of components in several products or variants provided they fit in the actual context.
- Stepwise refinement using feedback of operational data and the practice of creating forward-looking assurance cases (plans as to how to evolve the assurance case with the product) as part of planning product changes.
- Reusable patterns for arguments and contracts to promote simplicity and enable automation.
- Continuous assessment integrated with development and combined with confirmation reviews and audits.

The method is described in more detail in the paper "Continuous Deployment for Dependable Systems with Continuous Assurance Cases"[25]

## 6.8 Goal fulfilment

*Contribution to the goals of FFI and the programme for Road safety and automated vehicles*
While we have yet to see the impact of automated vehicles, they are expected to contribute to the FFI goals of reducing the environmental impact of road vehicles as well as reduce the number of fatalities and seriously injured due to road traffic. However, without solving the methodologic issues for safety assurance, it will be impossible to commercialize a self-driving vehicle[26]. This is also a prerequisite to reach the safety and automation goals of a predictive vehicle, as defined in the strategic roadmap for road safety and automated vehicles. This includes safety analysis of the division of responsibility between driver and vehicle. Hence the project fits well in contributing to the necessary areas of '*analysis, knowledge and enabling technology*' as well as '*basic safety properties*', as formulated in the road map.

*Goal fulfilment for project deliverables*
All deliverables in the project (except for the project website) were in the form of reports. The deliverables promised in the project application have been produced. These were:

- Report on use cases for autonomous personal cars (5 iterations)
- Report on use cases for commercial vehicles (5 iterations)
- Report on driver relations – state of the art (5 iterations)
- Report on driver relations – project results (4 iterations)
- Report on method for HARA – state of the art (5 iterations)
- Report on method for HARA – project results (4 iterations)
- Report on method for allocating safety requirements on decision elements – state of the art (5 iterations)
- Report on method for allocating safety requirements on decision elements – project results (4 iterations)
- Report on method for dividing safety integrity requirements when having incomplete redundancy – state of the art (5 iterations)
- Report on method for dividing safety integrity requirements when having incomplete redundancy – project results (4 iterations)
- Report on method for safety requirements refinement verification – state of the art (5 iterations)
- Report on method for safety requirements refinement verification – project results (4 iterations)

[25] *Warg, F., Blom, H., Borg, J., & Johansson, R., "Continuous Deployment for Dependable Systems with Continuous Assurance Cases," In: 2019 IEEE International Symposium on Software Reliability Engineering, WoSoCer workshop. IEEE Computer Society, 2019. (ESPLANADE paper)*
[26] *Markus Hörwick and Karl-Heinz Siedersberger, "Strategy and Architecture of a Safety Concept for Fully Automatic and Autonomous Driving Assistance Systems," IV, 2010.*

- Dissemination: Combined Plan and tracking report (5 iterations)
- Dissemination: Project website (https://esplanade-project.se)

*Goal fulfilment for project objective and research questions*
Methods were developed for all topic areas where initial research questions were formulated. In most cases the research questions were refined during the project, as anticipated when defining the iterative way of working. A short summary for each topic:

- In the area of driver relations, the initial research question was on what agreements are needed between ADS and human users in general and a simple way of categorizing agreements was developed. However, in the project the major focus became one specific but key agreement, the transfer of control between ADS and a human driver in a vehicle which may be on the move in traffic. Both a safety analysis method and design guidelines were developed to address this issue.
- For HARA, the developed method QRN answers three of the four initial research questions on ensuring completeness and usefulness of safety goals including for product lines. It also rendered the fourth question on finding hazardous events that create unique safety goals moot since the new method does not make use of hazardous events.
- In the topic of decision hierarchies and architecture patterns, some advances were done on several complementary methods: function analysis for architecture development mapping functions (including those related to decisions) to elements, a formal framework for reasoning on operational behaviour (related to run-time operational capability), and on using service-oriented architectures for greater flexibility.
- On the topic incomplete redundancy for sensor systems, several methods were also investigated: Probabilistic requirements, analysis of sensor limitations, and contract-based design for sensor safety requirements. While a complete solution to the research question has not been found, new questions were also defined for this complex issue.
- For refinement verification, the proposed method is using component-based design where components are developed independently with safety contracts and refinements is done with “the leaning V-model”. Additional work on the methodology is defined as future research.

As the project progressed, less emphasis was placed on evolving the use-cases as this was not deemed necessary to support the other activities. Hence the updates of the reports on use-cases were minor after the initial iteration. Less effort than planned was also spent on evaluation with prototype tools since the major method advancements came relatively late in the project giving less time for evaluation. However, promising methods were developed for all topic areas and a significant number of peer-reviewed scientific publications detailing many of them were written and presented at conferences or published in journals and promoted in invited talks, while some methods are instead detailed in the project deliverables. As expected, the project also resulted in new or refined research questions related to the topics. A summary of these can be found in Section 8.1 on future research.

# 7 Dissemination and publications

## 7.1 Dissemination

| How are the project results planned to be used and disseminated? | Mark with X | Comment |
|---|---|---|
| Increase knowledge in the field | X | Results disseminated through publications, presentations, seminars and internally in partner organizations. |
| Be passed on to other advanced technical development projects | X | Results to be used/refined in other research projects and partner-internal projects. |
| Be passed on to product development projects | X | At least part of the results is planned to be used in development of automated driving systems. |
| Introduced on the market | | |
| Used in investigations / regulatory / licensing / political decisions | X | Some results can be relevant for standardization. |

## 7.2 Publications

| Title | Author(s) | Conference/Journal |
|---|---|---|
| A Strategy for Assessing Safe Use of Sensors in Autonomous Road Vehicles | Rolf Johansson, Samieh Alissa, Staffan Bengtsson, Carl Bergenhem, Olof Bridal, Anders Cassel, DeJiu Chen, Martin Gassilewski, Jonas Nilsson, Anders Sandberg, Thomas Söderqvist, Stig Ursing, Fredrik Warg, Anders Werneman | SAFECOMP 2017 |
| A Model-based Approach to Dynamic Self-Assessment for Automated Performance and Safety Awareness of Cyber-Physical Systems | DeJiu Chen, Zhonghai Lu | IMBSA 2017 |
| Introducing ASIL Inspired Dynamic Tactical Safety Decision Framework for Automated Vehicles | Siddartha Khastgir, Anders Sandberg, Gunwant Dhadyalla, Håkan Sivencrona, Peter Billing, Stewart Birrell, Paul Jennings | ITSC 2017 |
| Safe Transitions Between a Driver and an Automated Driving System | Rolf Johansson, Jonas Nilsson, Annika Larsson | International Journal of Advances in Systems and Measurement, vol 10, numbers 3-4, 2017 |
| Signal Feature Analysis for Dynamic Anomaly Detection of Components in Embedded Control Systems | Xin Tao, DeJiu Chen, Juan Sagarduy | DepCoS 2018 |
| Design of a Knowledge-Base Strategy for Capability-Aware Treatment of Uncertainties of Automated Driving Systems | DeJiu Chen, Kenneth Östberg, Matthias Becker, Håkan Sivencrona, Fredrik Warg | WAISE 2018 |
| Enabling Tomorrow's Road Vehicles by Service-Oriented Platform Patterns | Rolf Johansson, Rikard Andersson, Markus Dernevik | European Congress Embedded Software and Real-time Systems, ERTS$^2$ 2018 |

| | | |
|---|---|---|
| A Business Model for selling components with safety certificates | Carl Bergenhem, Daniel Skarin, Fabian Wenger, Rolf Johansson | The Safety-Critical Systems Club, Seminar: COTS, Legacy and Reuse - poster session, 2018 |
| Argument Patterns for Multi-Concern Assurance of Connected Automated Driving Systems | Fredrik Warg, Martin Skoglund | The 4th International Workshop on Security and Dependability of Critical Embedded Real-Time Systems (CERTS 2019), July 2019. |
| Continuous Deployment for Dependable Systems with Continuous Assurance Cases | Fredrik Warg, Hans Blom, Jonas Borg, Rolf Johansson | The 9th IEEE International Workshop on Software Certification (WoSoCer 2019), October 2019. |
| Safer Transitions of Responsibility for Highly Automated Driving: Designing HMI for Transitions with Functional Safety in Mind | Matthew Sassman, Richard Wiik | The 10th European Congress on Embedded Real-Time Systems (ERTS 2020) |
| Towards an Operational Design Domain That Supports the Safety Argumentation of an Automated Driving System | Magnus Gyllenhammar, Rolf Johansson, Fredrik Warg, DeJiu Chen, Hans-Martin Heyn, Martin Sanfridson, Jan Söderberg, Anders Thorsén, Stig Ursing | The 10th European Congress on Embedded Real-Time Systems (ERTS 2020) |
| Towards Safety Analysis of Interactions Between Human Users and Automated Driving Systems | Fredrik Warg, Stig Ursing, Martin Kaalhus, Richard Wiik | The 10th European Congress on Embedded Real-Time Systems (ERTS 2020) - *Winner of best paper award in category 'Human System Interactions'* |
| Defining Fundamental Vehicle Actions for the Development of Automated Driving Systems | Magnus Gyllenhammar, Carl Zandén | SAE World Congress Experience 2020 (WCX 2020) |
| On Perception Safety Requirements and Multi Sensor Systems for Automated Driving Systems | Anders Cassel, Carl Bergenhem, Ole Martin Christensen, Hans-Martin Heyn, Susanna Leadersson-Olsson, Mario Majdandzic, Peng Sun, Anders Thorsén, Jörgen Tryggvesson | SAE World Congress Experience 2020 (WCX 2020) |
| The Quantitative Risk Norm - A Proposed Tailoring of HARA for ADS | Fredrik Warg, Rolf Johansson, Martin Skoglund, Anders Thorsén, Mattias Brännström, Magnus Gyllenhammar, Martin Sanfridson | 2020 Workshop on safety and security of intelligent vehicles (SSIV 2020) |

Two further papers were written near the end of the project but are, at the time of writing, not yet published. Tentative titles are: *"The Frequency-based Operational Design Domain and the Role of Minimal Risk Condition for Safe Automated Driving Systems"* (Magnus Gyllenhammar, Mattias Brännström, Rolf Johansson, Fredrik Sandblom, Martin Sanfridson, Martin Törngren, Stig Ursing and Fredrik Warg) and *"Concepts and Risk Analysis for a Cooperative and Automated Highway Vehicle System"* (Carl Bergenhem, Mario Majdandzic and Stig Ursing).

# 8 Conclusions and future research

The development of automated driving systems has seen major investments in recent years. The hopes are that these systems will contribute to a future of more efficient, accessible, and safer transport solutions, which is of great importance for many reasons. During the time the

project has been running, the realization that being able to show that an ADS is safe is a key hurdle when it comes to automated vehicles has increased in the automotive domain.

The question of how to perform safety assurance of automated vehicles is large and the focus of many research initiatives. Any single project will not solve all open questions, but to our own assessment ESPLANADE has both helped advance state-of-the-art and provided useful directions for future research in the area. The gained knowledge has been shared through 16 scientific papers and a number of invited talks and seminars. This report has summarized the major findings in the project.

## 8.1 Future research

A number of directions for future research in safety assurance for ADS have been identified:

- *Continuous deployment agreements*: In order to enable the possibility of adding features to a vehicle in the field the agreements must be clearly specified. How to ensure safety in adding new features needs further investigation, but a conceptual picture can be seen in Figure 19.
- *Minimal Risk Condition (MRC)*: Achieving a safe state during possible faults in transitions or in operation is vital to the safety case argument. How to properly define the MRC and also the relation to human users is still an unsolved question.
- Using quantitative safety goals in the QRN approach instead of the discrete integrity levels of ISO 26262 indicates moving towards a quantitative assurance framework altogether may be good idea. How this may look, and how to integrate methods based on a qualitative judgment in the framework is future work.
- Creating a functional safety concept based on QRN safety goals, and verification and validation of such QRN safety goals.
- Use of frequency-based ODD parameters and how to handle current risk in run-time to fulfil the safety goals.
- How to represent ODDs for exchanging information and how to evolve ODD definitions with feature updates.
- Establishing a solid mathematical description of incomplete redundancy and the contract-based design of the perception system.
- Definition and development of sensor and capability model and in what way to define the capability in design and run-time.
- Methods for determination and specification of sufficient redundancy, separation of non-deterministic functionality and corresponding safety architecture design patterns.
- Continued work on how to ensure safety in the entire life cycle in order to support modularity and continuous deployment. This includes an efficient way to formulate semi-formal safety contracts for use in requirements refinement.
- The enrichment of uncertainty modelling for the analysis of safety knowledge for ADS as well as the synthesis of safety rules.
- Finally, the issue of how to define acceptable risk and the relationship to legal consequences would be necessary in order to define a risk norm in the QRN approach.

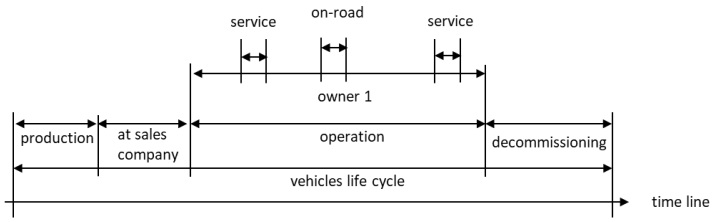


*Figure 19: Continuous deployment agreements.*

# 9 Participating parties and persons

List of contact persons for participating organizations:

| Name | Email | Partner Affiliation |
|---|---|---|
| Fredrik Warg (project coordinator) | fredrik.warg@ri.se | RISE Research Institutes of Sweden |
| Murat Erdogan | murat.erdogan@veoneer.com | Veoneer |
| Ola Örsmark | ola.orsmark@comentor.se | Comentor |
| Anders Sandberg | anders.sandberg@aptiv.com | Aptiv |
| De-Jiu Chen | chen@md.kth.se | KTH Kungliga Tekniska Högskolan |
| Stig Ursing | stig.ursing@semcon.com | Semcon Sweden |
| Jan Söderberg | jan.soderberg@systemite.se | Systemite |
| Thomas Söderqvist | thomas.soderqvist@volvo.com | Volvo Group |
| Carl Bergenhem | carl.bergenhem@qamcom.se | Qamcom Research and Technology |
| Magnus Gyllenhammar | magnus.gyllenhammar@zenuity.com | Zenuity |